



Как устроена разработка Kubernetes-платформы Deckhouse

Константин Аксёнов





Обо мне



Константин Аксёнов

Руководитель разработки
Deckhouse Kubernetes Platform

konstantin.aksenov@flant.com

github.com/konstantin-axenov

Чем занимаюсь

Больше 5 лет засыпаю и просыпаюсь с мыслями о Kubernetes.

Опыт

С 2011 занимаюсь разработкой.

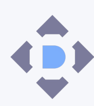
С 2017 работаю в компании «Флант».

С 2020 руководитель разработки Deckhouse Kubernetes Platform.

С чем работаю больше всего



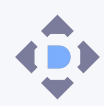




Доклад будет интересен тем, кто



Использует Kubernetes в production



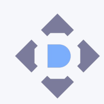
Доклад будет интересен тем, кто



Использует Kubernetes в production



Только присматривается к Kubernetes



Доклад будет интересен тем, кто



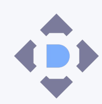
Использует Kubernetes в production



Только присматривается к Kubernetes



Задумывается над разработкой своего «аналога OpenShift»



Доклад будет интересен тем, кто



Использует Kubernetes в production



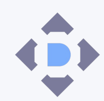
Только присматривается к Kubernetes



Задумывается над разработкой своего «аналога OpenShift»

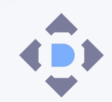


Просто любит Open Source



«Работать с Kubernetes умеет каждый»

Что вы, вообще, сделали?
Взяли Kubernetes и засунули в него разные Open Source-компоненты?

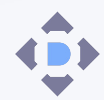


«Работать с Kubernetes умеет каждый»



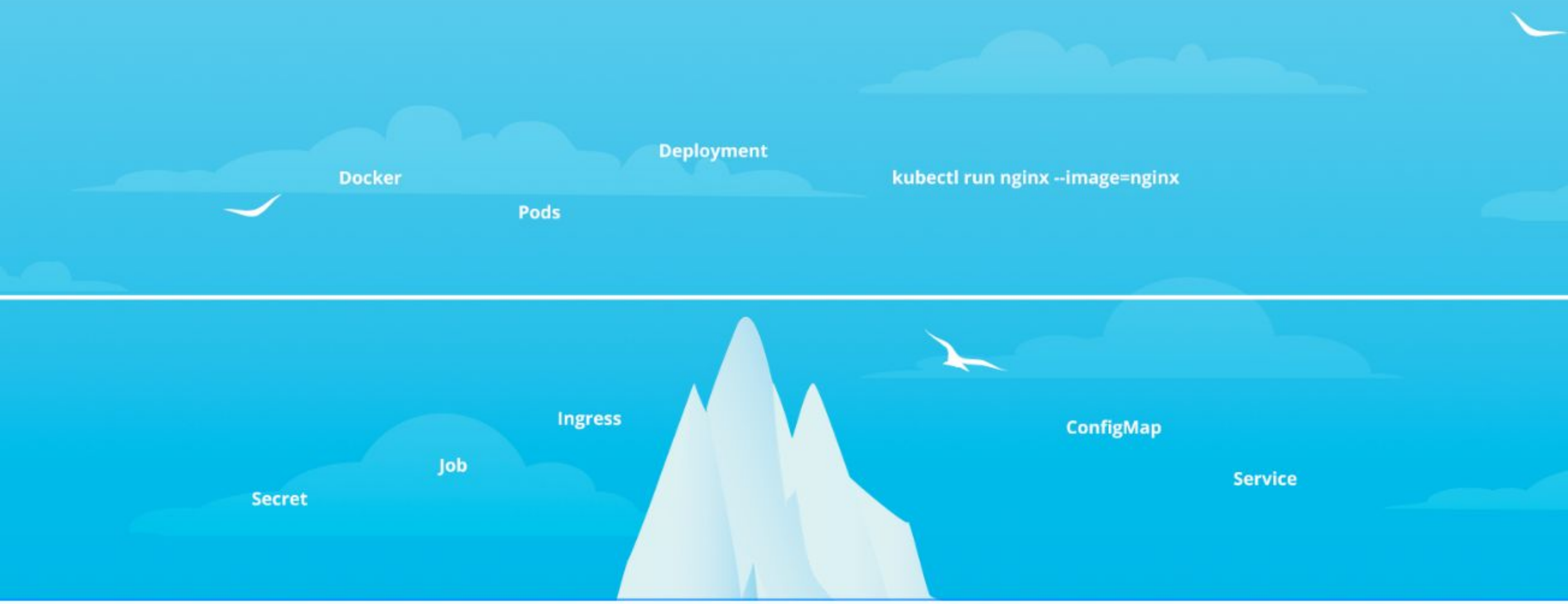
Что вы, вообще, сделали?
Взяли Kubernetes и засунули в него разные Open Source-компоненты?





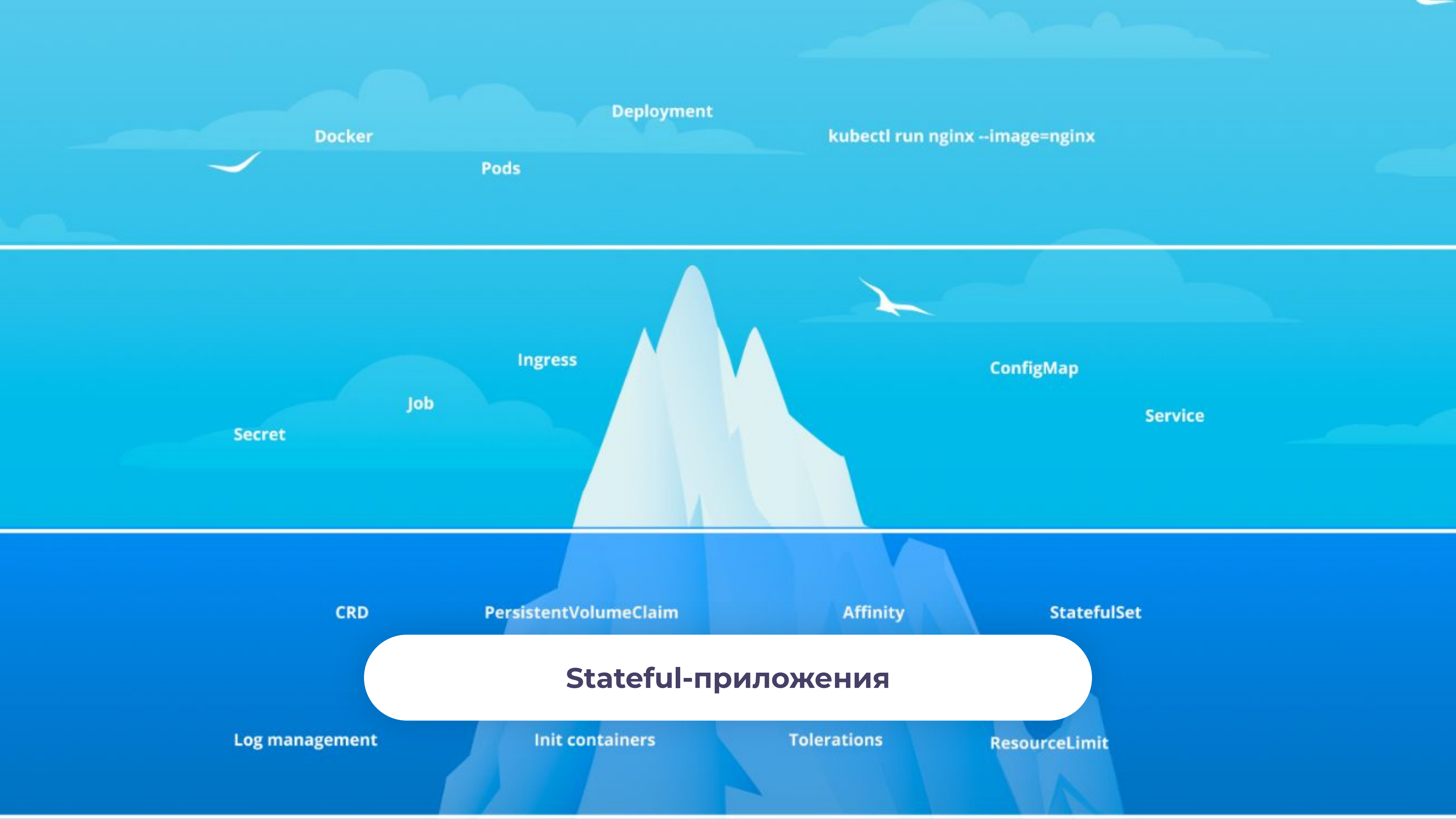
Kubernetes — это не только YAML

Kubernetes — это не только YAML

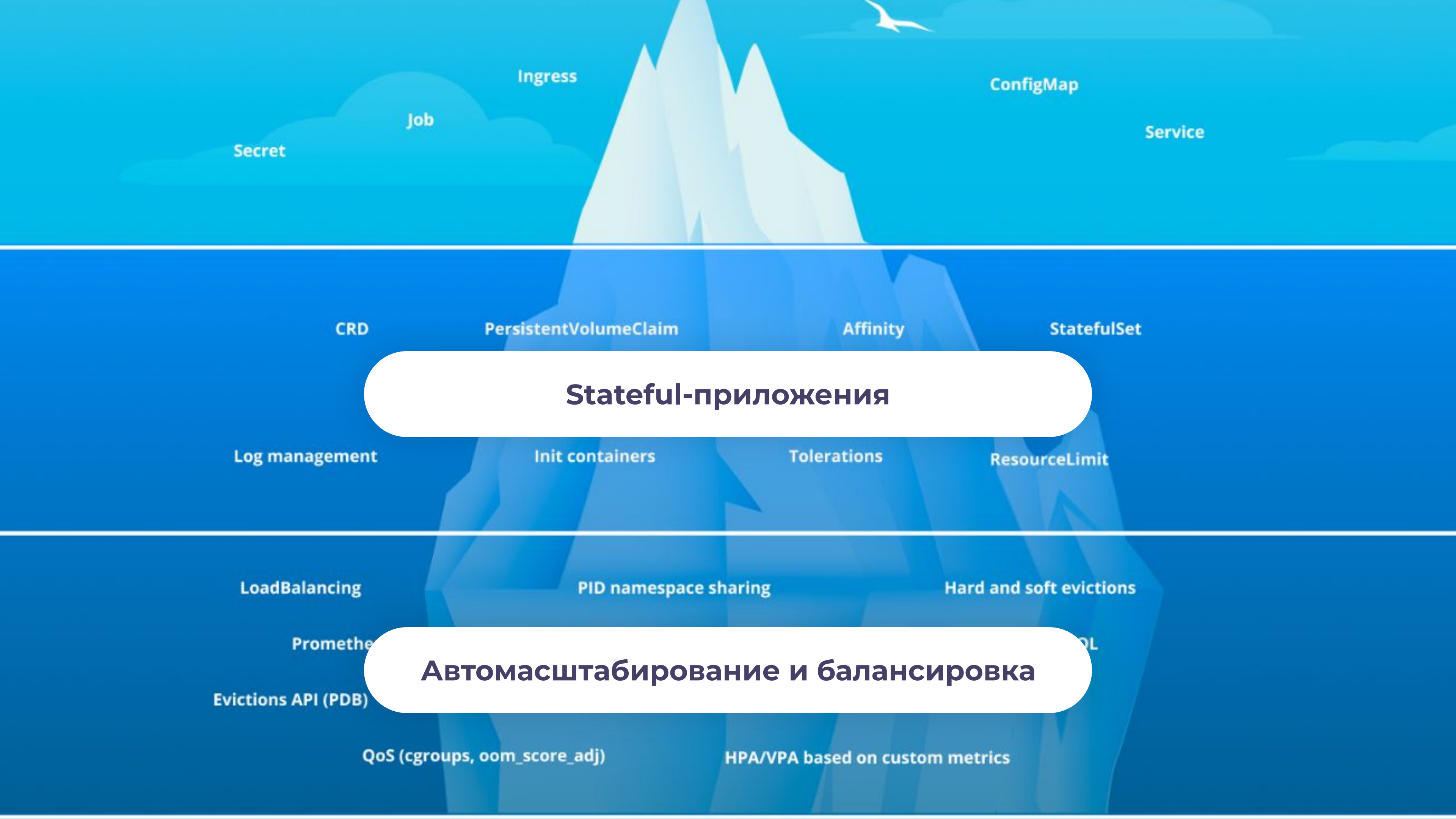


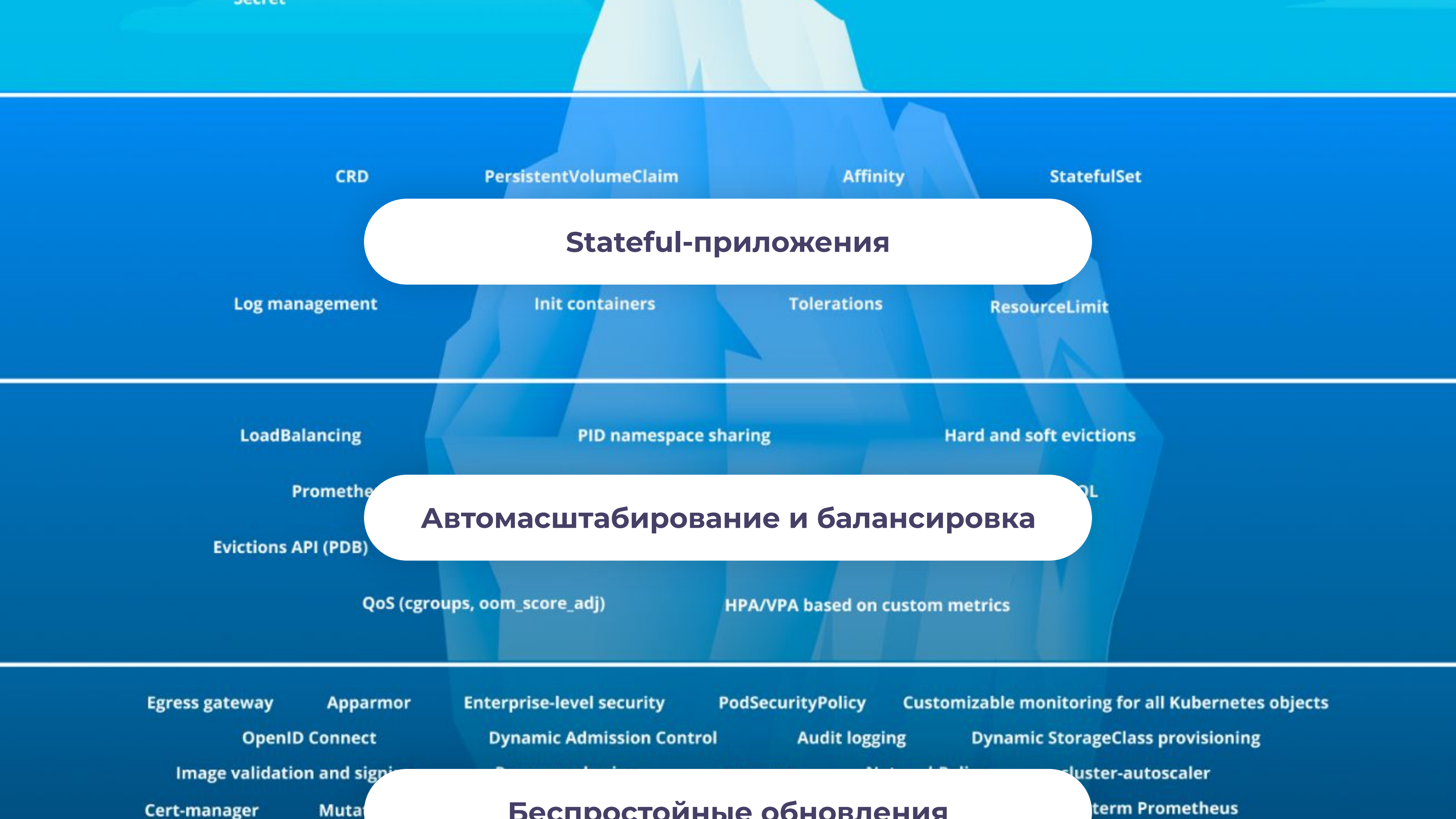
Kubernetes — это не только YAML

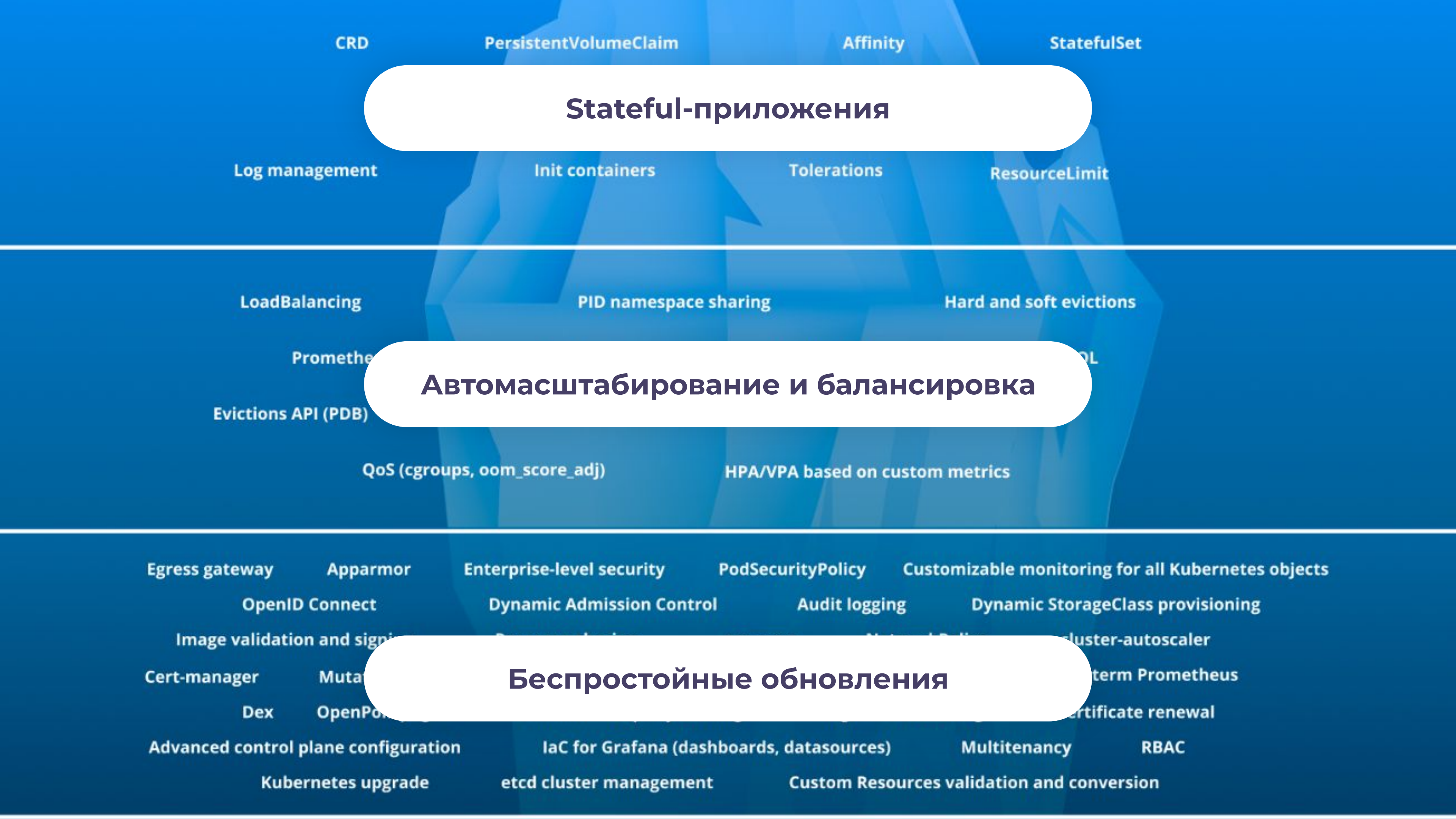



Docker
Deployment
Pods
kubectl run nginx --image=nginx
Ingress
Job
Secret
ConfigMap
Service
CRD
PersistentVolumeClaim
Affinity
StatefulSet
Stateful-приложения
Log management
Init containers
Tolerations
ResourceLimit

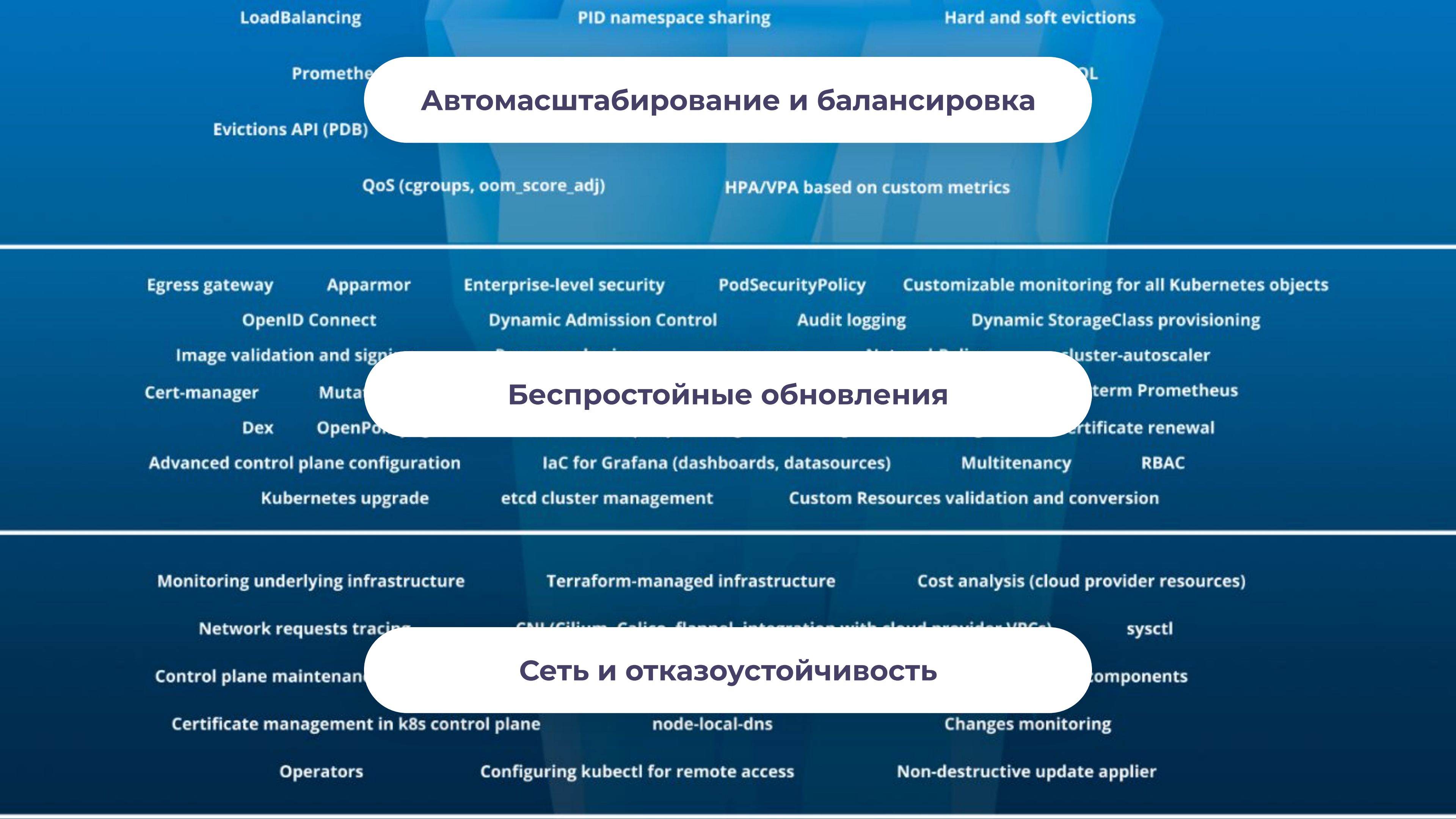

Pods
Ingress
Job
Secret
ConfigMap
Service
CRD
PersistentVolumeClaim
Affinity
StatefulSet
Stateful-приложения
Log management
Init containers
Tolerations
ResourceLimit
LoadBalancing
PID namespace sharing
Hard and soft evictions

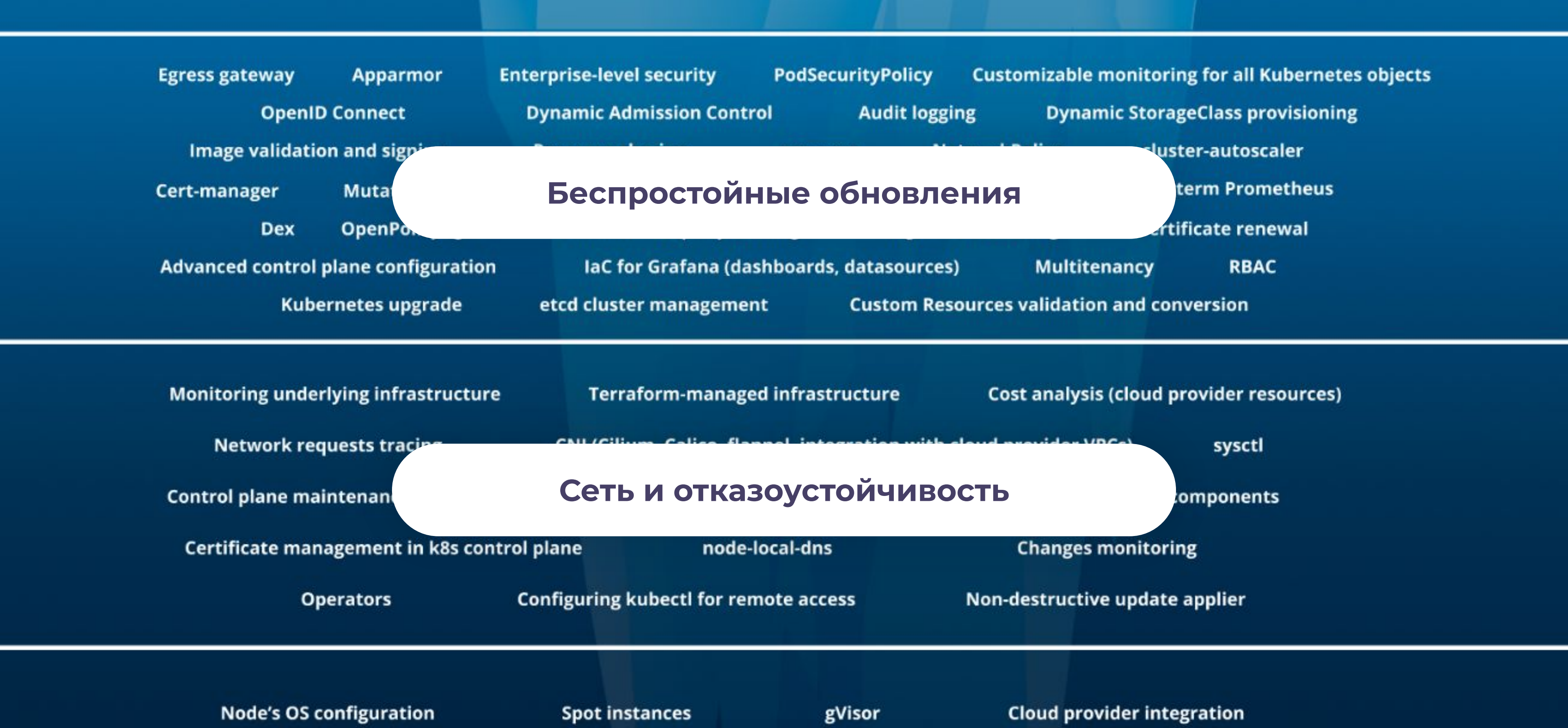

Ingress
ConfigMap
Job
Service
Secret
CRD
PersistentVolumeClaim
Affinity
StatefulSet
Stateful-приложения
Log management
Init containers
Tolerations
ResourceLimit
LoadBalancing
PID namespace sharing
Hard and soft evictions
Автомасштабирование и балансировка
Evictions API (PDB)
QoS (cgroups, oom_score_adj)
HPA/VPA based on custom metrics


CRD
PersistentVolumeClaim
Affinity
StatefulSet
Stateful-приложения
Log management
Init containers
Tolerations
ResourceLimit
LoadBalancing
PID namespace sharing
Hard and soft evictions
Automасштабирование и балансировка
Evictions API (PDB)
QoS (cgroups, oom_score_adj)
HPA/VPA based on custom metrics
Egress gateway
Apparmor
Enterprise-level security
PodSecurityPolicy
Customizable monitoring for all Kubernetes objects
OpenID Connect
Dynamic Admission Control
Audit logging
Dynamic StorageClass provisioning
Cert-manager
Беспростойные обновления


CRD
PersistentVolumeClaim
Affinity
StatefulSet
Stateful-приложения
Log management
Init containers
Tolerations
ResourceLimit
LoadBalancing
PID namespace sharing
Hard and soft evictions
Автомасштабирование и балансировка
Evictions API (PDB)
QoS (cgroups, oom_score_adj)
HPA/VPA based on custom metrics
Egress gateway
Apparmor
Enterprise-level security
PodSecurityPolicy
Customizable monitoring for all Kubernetes objects
OpenID Connect
Dynamic Admission Control
Audit logging
Dynamic StorageClass provisioning
Cert-manager
Dex
Беспростойные обновления
Advanced control plane configuration
IaC for Grafana (dashboards, datasources)
Multitenancy
RBAC
Kubernetes upgrade
etcd cluster management
Custom Resources validation and conversion


Log management
Init containers
Tolerations
ResourceLimit
LoadBalancing
PID namespace sharing
Hard and soft evictions
Evictions API (PDB)
QoS (cgroups, oom_score_adj)
HPA/VPA based on custom metrics
Автомасштабирование и балансировка
Egress gateway
Apparmor
Enterprise-level security
PodSecurityPolicy
Customizable monitoring for all Kubernetes objects
OpenID Connect
Dynamic Admission Control
Audit logging
Dynamic StorageClass provisioning
Cert-manager
Dex
Advanced control plane configuration
IaC for Grafana (dashboards, datasources)
Multitenancy
RBAC
Kubernetes upgrade
etcd cluster management
Custom Resources validation and conversion
Беспростойные обновления
Monitoring underlying infrastructure
Terraform-managed infrastructure
Cost analysis (cloud provider resources)
sysctl


LoadBalancing
PID namespace sharing
Hard and soft evictions
Evictions API (PDB)
Автомасштабирование и балансировка
QoS (cgroups, oom_score_adj)
HPA/VPA based on custom metrics
Egress gateway
Apparmor
Enterprise-level security
PodSecurityPolicy
Customizable monitoring for all Kubernetes objects
OpenID Connect
Dynamic Admission Control
Audit logging
Dynamic StorageClass provisioning
Cert-manager
Беспростойные обновления
Dex
Advanced control plane configuration
IaC for Grafana (dashboards, datasources)
Multitenancy
RBAC
Kubernetes upgrade
etcd cluster management
Custom Resources validation and conversion
Monitoring underlying infrastructure
Terraform-managed infrastructure
Cost analysis (cloud provider resources)
sysctl
Сеть и отказоустойчивость
Certificate management in k8s control plane
node-local-dns
Changes monitoring
Operators
Configuring kubectl for remote access
Non-destructive update applier

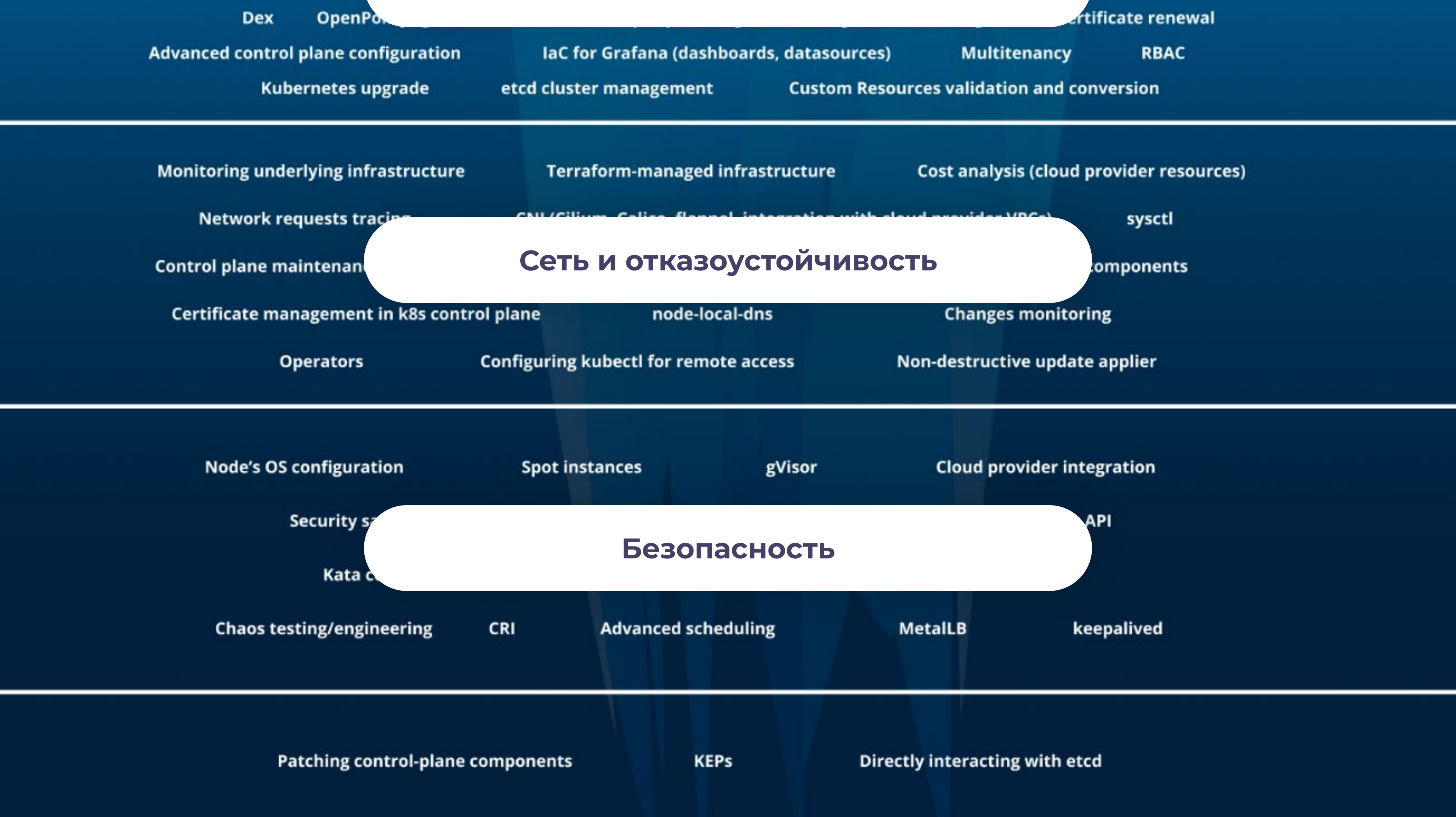


Беспростойные обновления

Сеть и отказоустойчивость

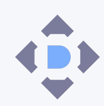
Безопасность


Egress gateway
AppArmor
Enterprise-level security
PodSecurityPolicy
Customizable monitoring for all Kubernetes objects
OpenID Connect
Dynamic Admission Control
Audit logging
Dynamic StorageClass provisioning
Image validation and sign
luster-autoscaler
Cert-manager
Muta
term Prometheus
Dex
OpenPo
rtificate renewal
Беспростойные обновления
Advanced control plane configuration
IaC for Grafana (dashboards, datasources)
Multitenancy
RBAC
Kubernetes upgrade
etcd cluster management
Custom Resources validation and conversion
Monitoring underlying infrastructure
Terraform-managed infrastructure
Cost analysis (cloud provider resources)
Network requests tracing
sysctl
Control plane maintenan
omponents
Сеть и отказоустойчивость
Certificate management in k8s control plane
node-local-dns
Changes monitoring
Operators
Configuring kubectl for remote access
Non-destructive update applier
Node’s OS configuration
Spot instances
gVisor
Cloud provider integration
Security s
API
Kata c
Безопасность
Chaos testing/engineering
CRI
Advanced scheduling
MetalLB
keepalived

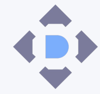

Dex
OpenPo
rtificate renewal
Advanced control plane configuration
IaC for Grafana (dashboards, datasources)
Multitenancy
RBAC
Kubernetes upgrade
etcd cluster management
Custom Resources validation and conversion
Monitoring underlying infrastructure
Terraform-managed infrastructure
Cost analysis (cloud provider resources)
Network requests tracing
sysctl
Сеть и отказоустойчивость
Control plane maintenan
omponents
Certificate management in k8s control plane
node-local-dns
Changes monitoring
Operators
Configuring kubectl for remote access
Non-destructive update applier
Node's OS configuration
Spot instances
gVisor
Cloud provider integration
Security s
API
Безопасность
Kata c
Chaos testing/engineering
CRI
Advanced scheduling
MetalLB
keepalived
Patching control-plane components
KEPs
Directly interacting with etcd

Требования к Kubernetes-платформе

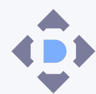
- Надежность
- Небольшая команда эксплуатации
- Идентичные Kubernetes-кластеры в любой инфраструктуре
- Снижение нагрузки на пользователей платформы — NoOps
- Безопасность



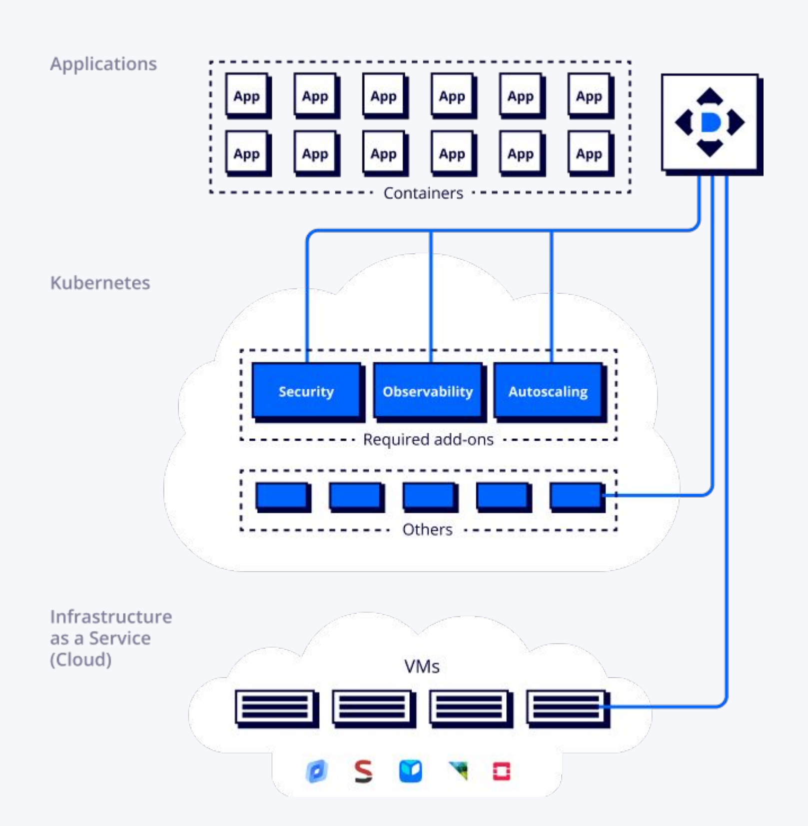
Поэтому мы сделали **Deckhouse** для себя



Поэтому мы сделали Deckhouse ~~для себя~~









Управляет
конфигурацией кластера и всех компонентов



Устанавливает
компоненты в кластер

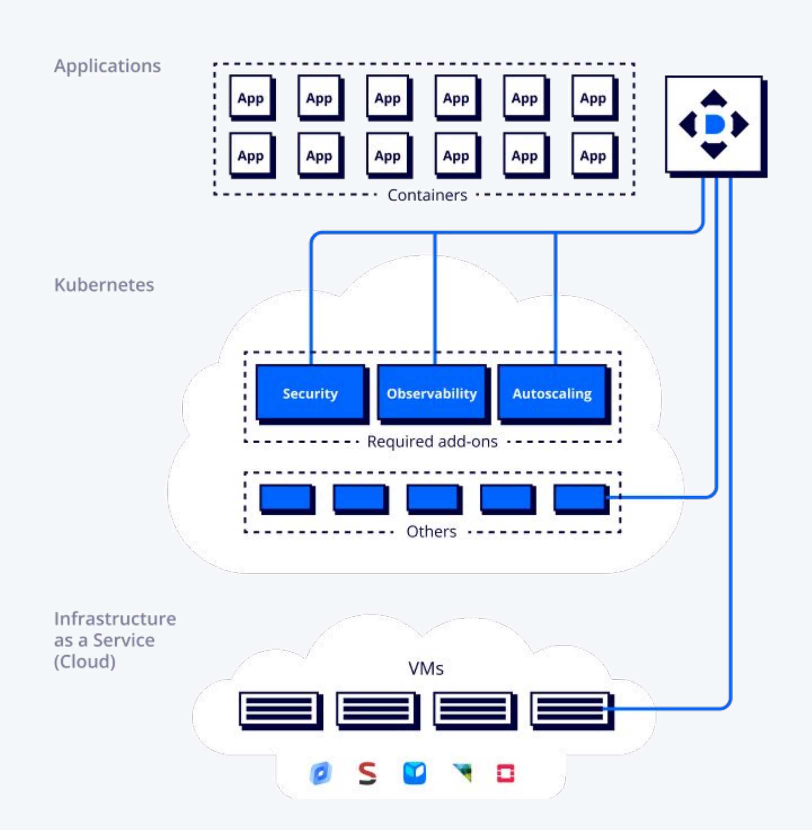


Разворачивает
Kubernetes кластер



Подготавливает
инфраструктуру или работает поверх bare-metal







Управляет
конфигурацией кластера и всех компонентов



Устанавливает
компоненты в кластер

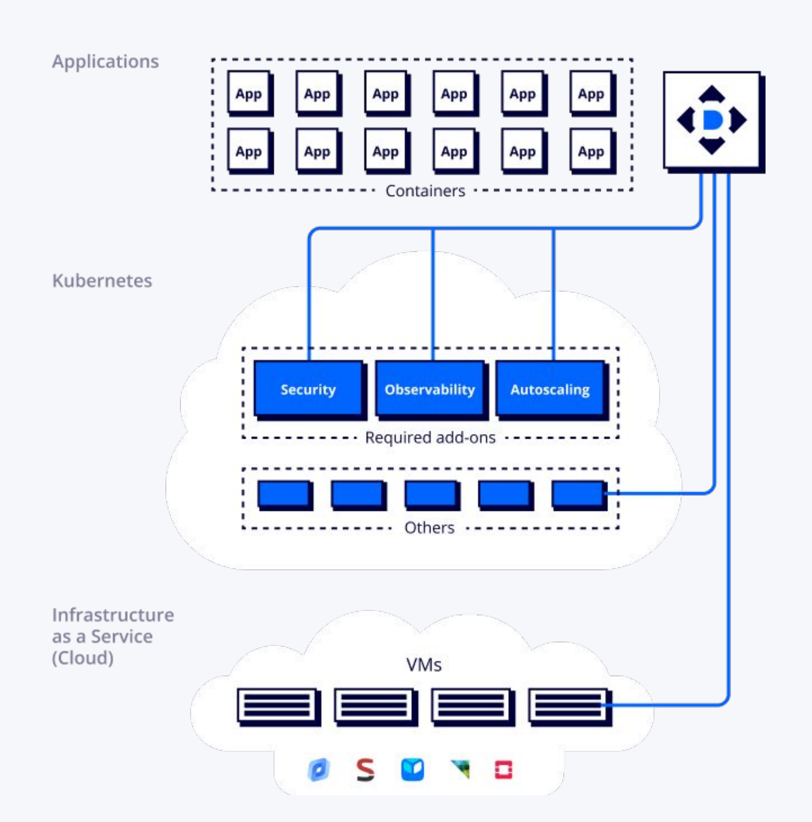


Разворачивает
Kubernetes-кластер



Подготавливает
инфраструктуру или работает поверх bare-metal







Управляет

конфигурацией кластера и всех компонентов



Устанавливает

компоненты в кластер



Разворачивает

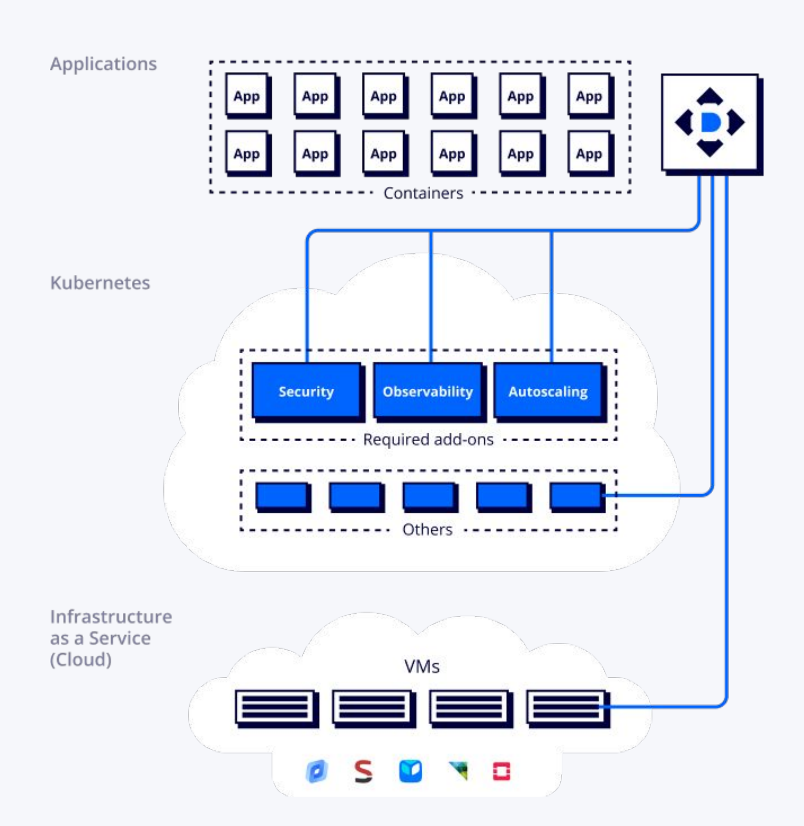
Kubernetes-кластер



Подготавливает

инфраструктуру или работает поверх bare-metal







Управляет

конфигурацией кластера и всех компонентов



Устанавливает

компоненты в кластер



Разворачивает

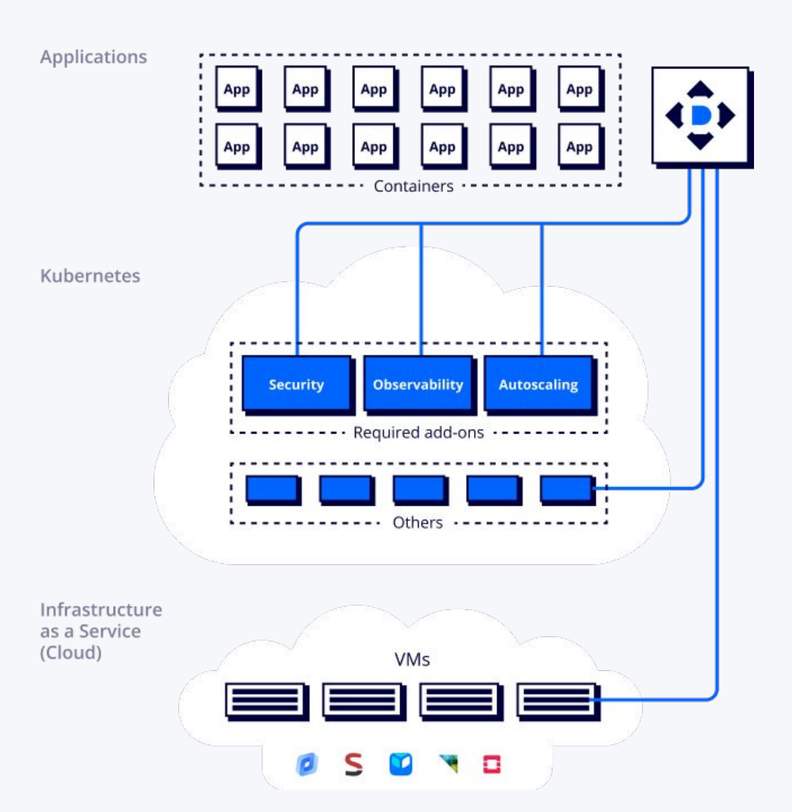
Kubernetes-кластер



Подготавливает

инфраструктуру или работает поверх Bare metal





Решает проблему айсберга

Docker
Deployment
kubectl run nginx --image=nginx
Pods

Ingress
Job
Secret
ConfigMap
Service

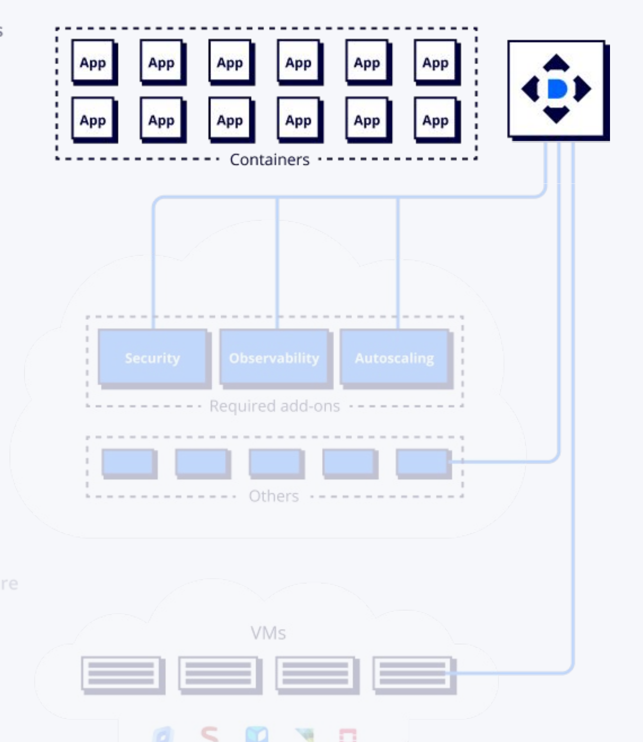
CRD
PersistentVolumeClaim
Affinity
StatefulSet
Helm templating
Dashboard
HPA
Log management
Init containers
Tolerations
ResourceLimit

LoadBalancing
PID namespace sharing
Hard and soft evictions
Prometheus
Grafana
Healthchecks
PromQL
Evictions API (PDB)
Terraform
Service discovery
QoS (cgroups, oom_score_adj)
HPA/VPA based on custom metrics

Egress gateway
Apparmor
Enterprise-level security
PodSecurityPolicy
Customizable monitoring for all Kubernetes objects
OpenID Connect
Dynamic Admission Control
Audit logging
Dynamic StorageClass provisioning
Image validation and signing
Resource sharing
seccomp
NetworkPolicy
cluster-autoscaler
Cert-manager
Mutating and Admission webhooks
Ingress monitoring
descheduler
Longterm Prometheus
Dex
OpenPolicyAgent
Prometheus query caching
Ingress autoscaling
Certificate renewal
Advanced control plane configuration
IaC for Grafana (dashboards, datasources)
Multitenancy
RBAC
Kubernetes upgrade
etcd cluster management
Custom Resources validation and conversion

Monitoring underlying infrastructure
Terraform-managed infrastructure
Cost analysis (cloud provider resources)
Network requests tracing
CNI (Cilium, Calico, flannel, integration with cloud provider VPCs)
sysctl





Решает проблему айсберга

Docker
Pods
Deployment
kubectl run nginx --image=nginx

Secret
Job
Ingress
ConfigMap
Service

CRD
PersistentVolumeClaim
Affinity
StatefulSet
Helm templating
Dashboard
HPA
Log management
Init containers
Tolerations
ResourceLimit

LoadBalancing
PID namespace sharing
Hard and soft evictions
Prometheus
Grafana
Healthchecks
PromQL
Evictions API (PDB)
Terraform
Service discovery
QoS (cgroups, oom_score_adj)
HPA/VPA based on custom metrics

Egress gateway
Apparmor
Enterprise-level security
PodSecurityPolicy
Customizable monitoring for all Kubernetes objects
OpenID Connect
Dynamic Admission Control
Audit logging
Dynamic StorageClass provisioning
Image validation and signing
Resource sharing
seccomp
NetworkPolicy
cluster-autoscaler
Cert-manager
Mutating and Admission webhooks
Ingress monitoring
descheduler
Longterm Prometheus
Dex
OpenPolicyAgent
Prometheus query caching
Ingress autoscaling
Certificate renewal
Advanced control plane configuration
IaC for Grafana (dashboards, datasources)
Multitenancy
RBAC
Kubernetes upgrade
etcd cluster management
Custom Resources validation and conversion

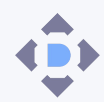
Monitoring underlying infrastructure
Terraform-managed infrastructure
Cost analysis (cloud provider resources)
Network requests tracing
CNI (Cilium, Calico, flannel, integration with cloud provider VPCs)
sysctl

Продукт с историей

29 сентября 2017 — Первый коммит

Июль 2020 — Enterprise Edition (EE)

28 июля 2021 — Community Edition (CE)



Команда



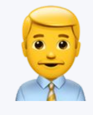
Product Manager



Release Engineer



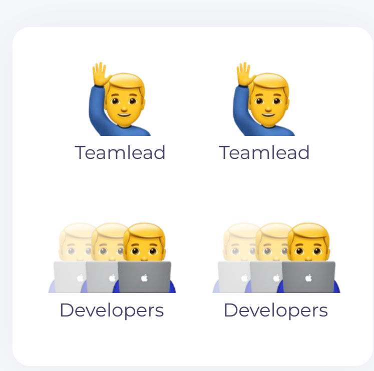
Tech Lead/Architect



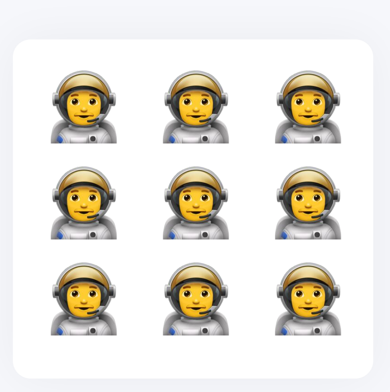
Account Manager



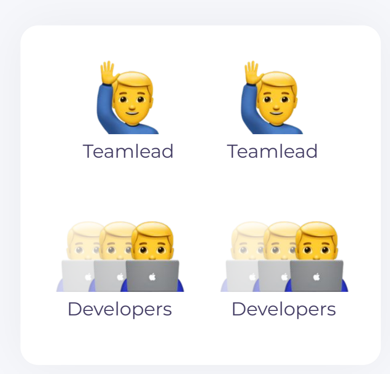
Engineering Manager



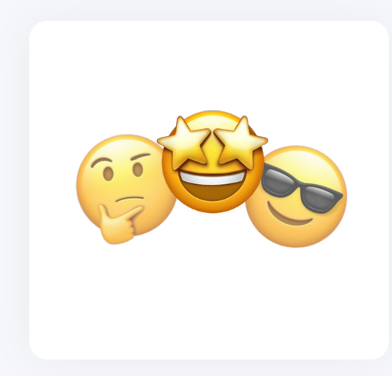
Dev Team



Flant DevOps Engineers



Dev Team

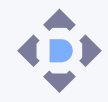


Community



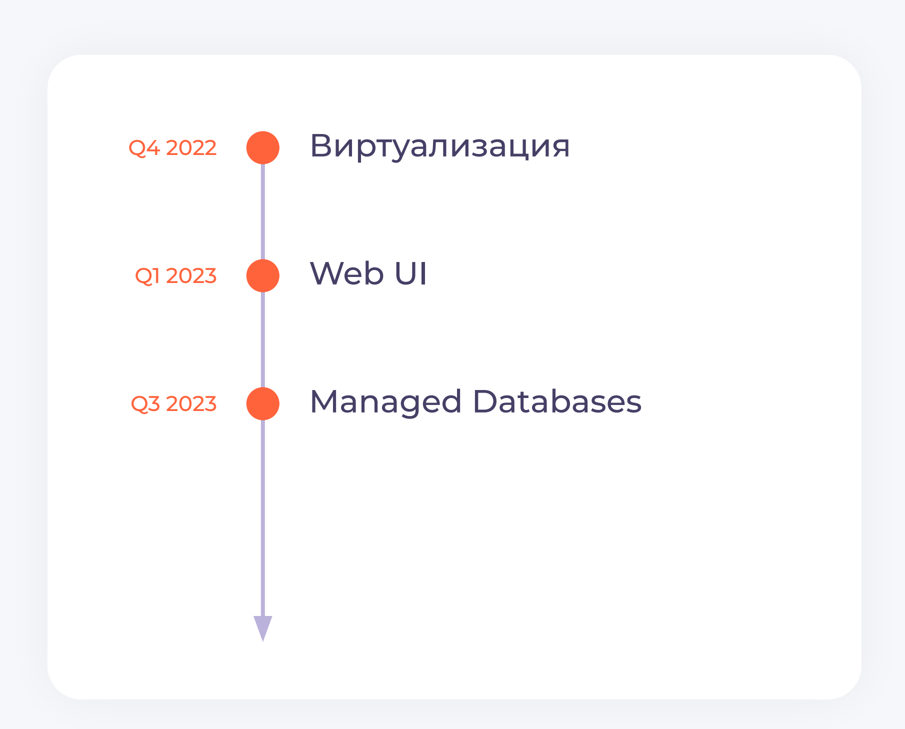
Жизненный цикл

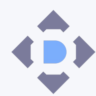
code
plan
build
Dev
release
test
deploy
Ops
operate
monitor



Планирование: источники задач

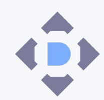
- Product owner
  - Сформирован roadmap до 2025 года





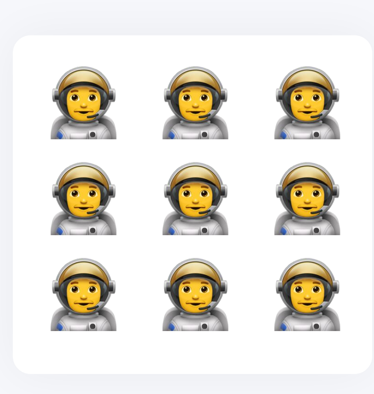
Планирование: источники задач

- Product owner
  - Сформирован roadmap до 2025 года
- Команда Deckhouse
  - Фиксы багов, работа над стабильностью
  - CI и тесты
  - Технический долг (переписали с Bash на Go)
  - Обновления Kubernetes и компонентов

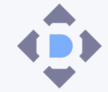


Планирование: источники задач

- Product owner
  - Сформирован roadmap до 2025 года
- Команда Deckhouse
  - Фиксы багов, работа над стабильностью
  - CI и тесты
  - Технический долг (переписали с Bash на Go)
  - Обновления Kubernetes и компонентов
- Пользователи Managed K8s и Enterprise-версии
  - Баги
  - Новая функциональность
  - Улучшения документации

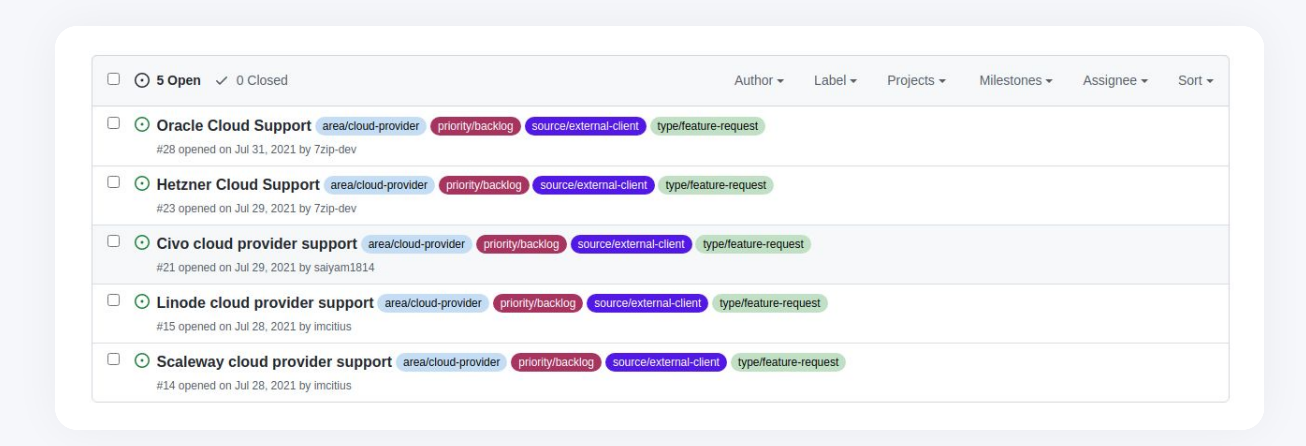


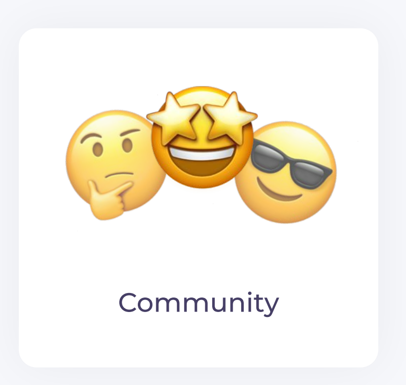
DevOps Engineers



Планирование: источники задач

- Community-пользователи и большие запросы
  - Поддержка всех возможных провайдеров
  - Поддержка всех возможных ОС

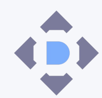






Планирование: сортировка задач

- Периодичность
  - Один раз в неделю
- Обработка
  - Источник
  - Тип
  - Раздел Deckhouse
  - Приоритет

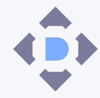


Планирование: сортировка задач

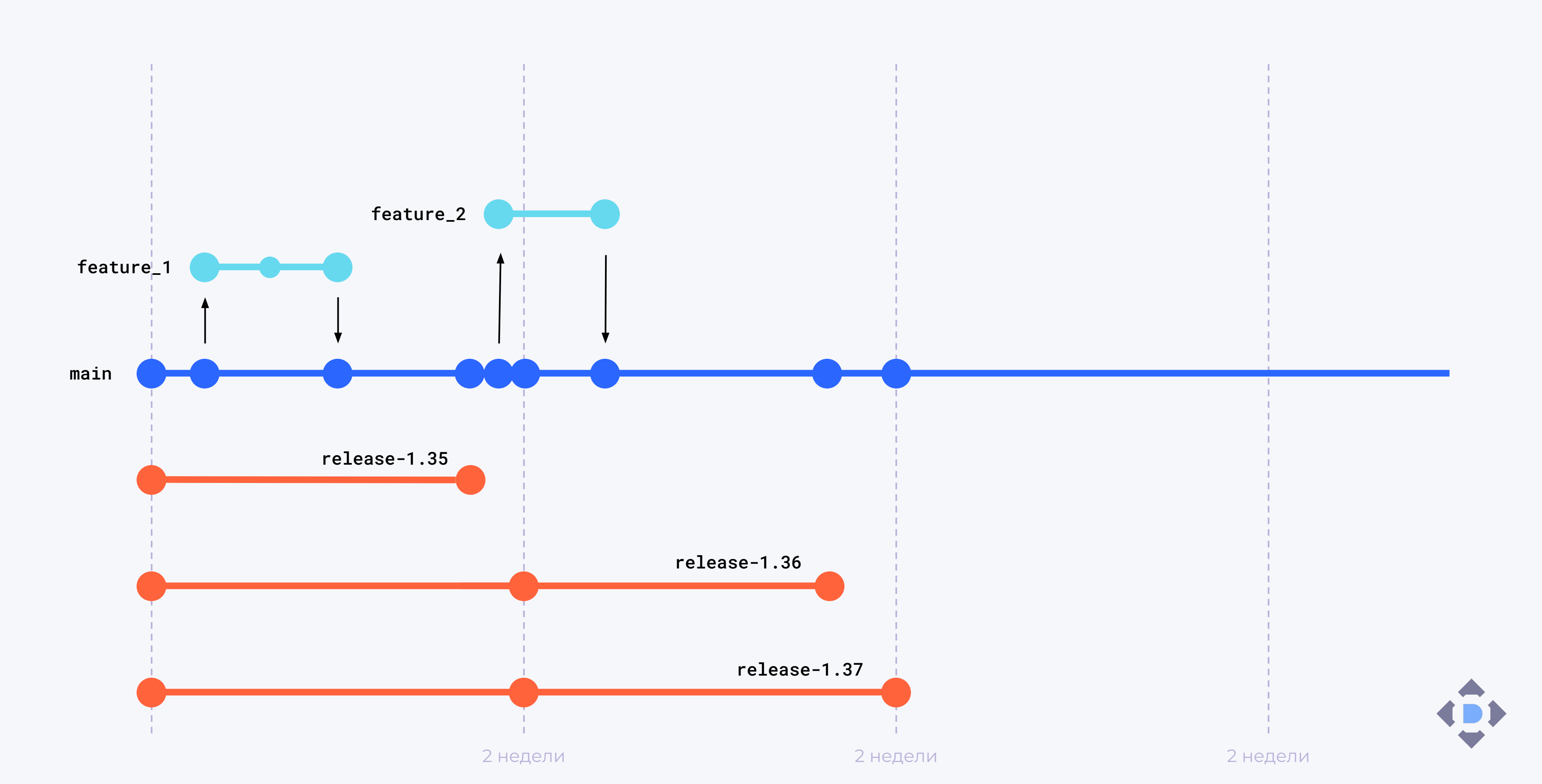
- Периодичность
  - Один раз в неделю
- Обработка
  - Источник
  - Тип
  - Раздел Deckhouse
  - Приоритет

Планирование: релиз

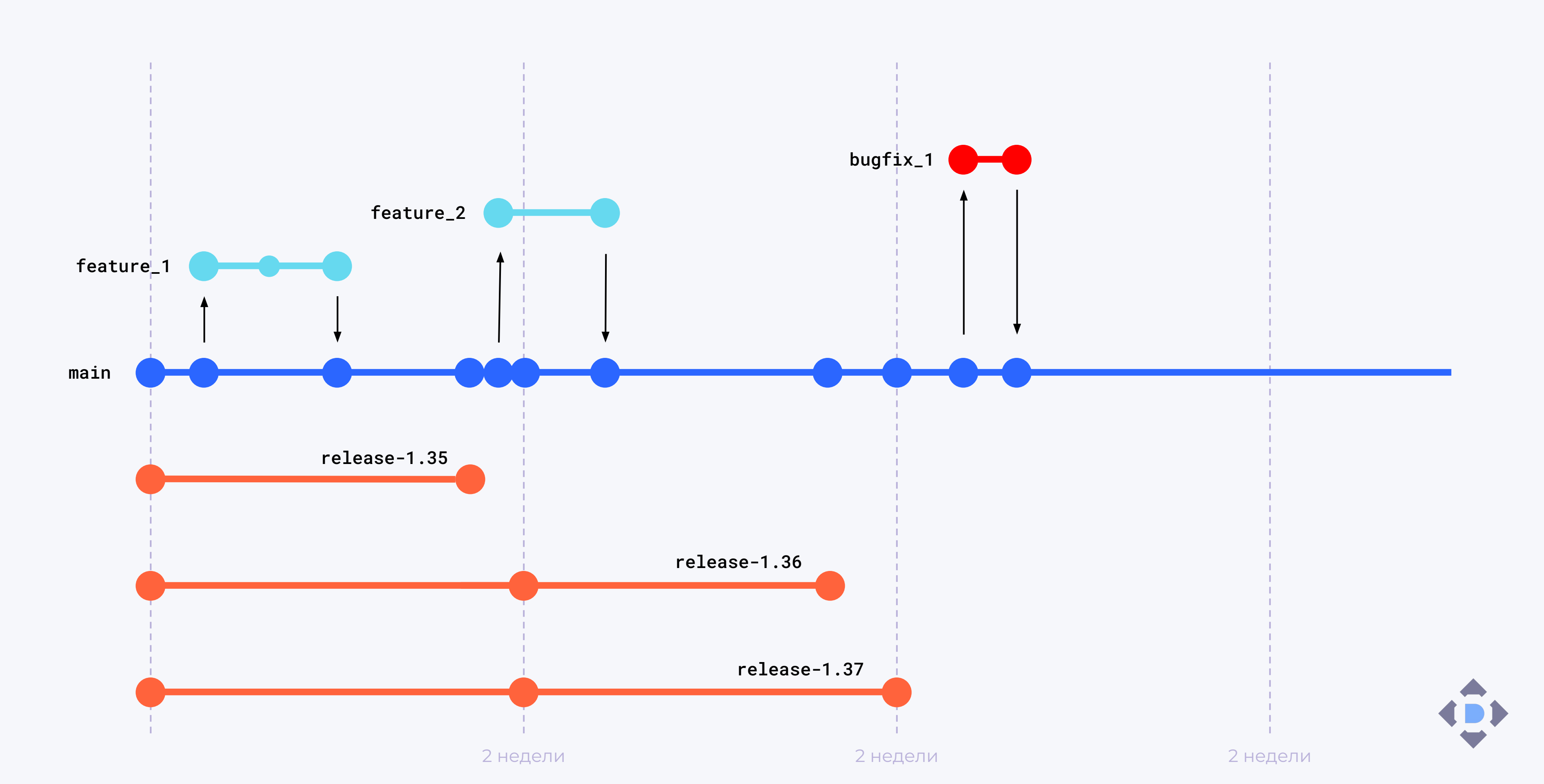
- Минорные релизы
  - Один раз в две недели
- Патч релизы
  - По мере необходимости



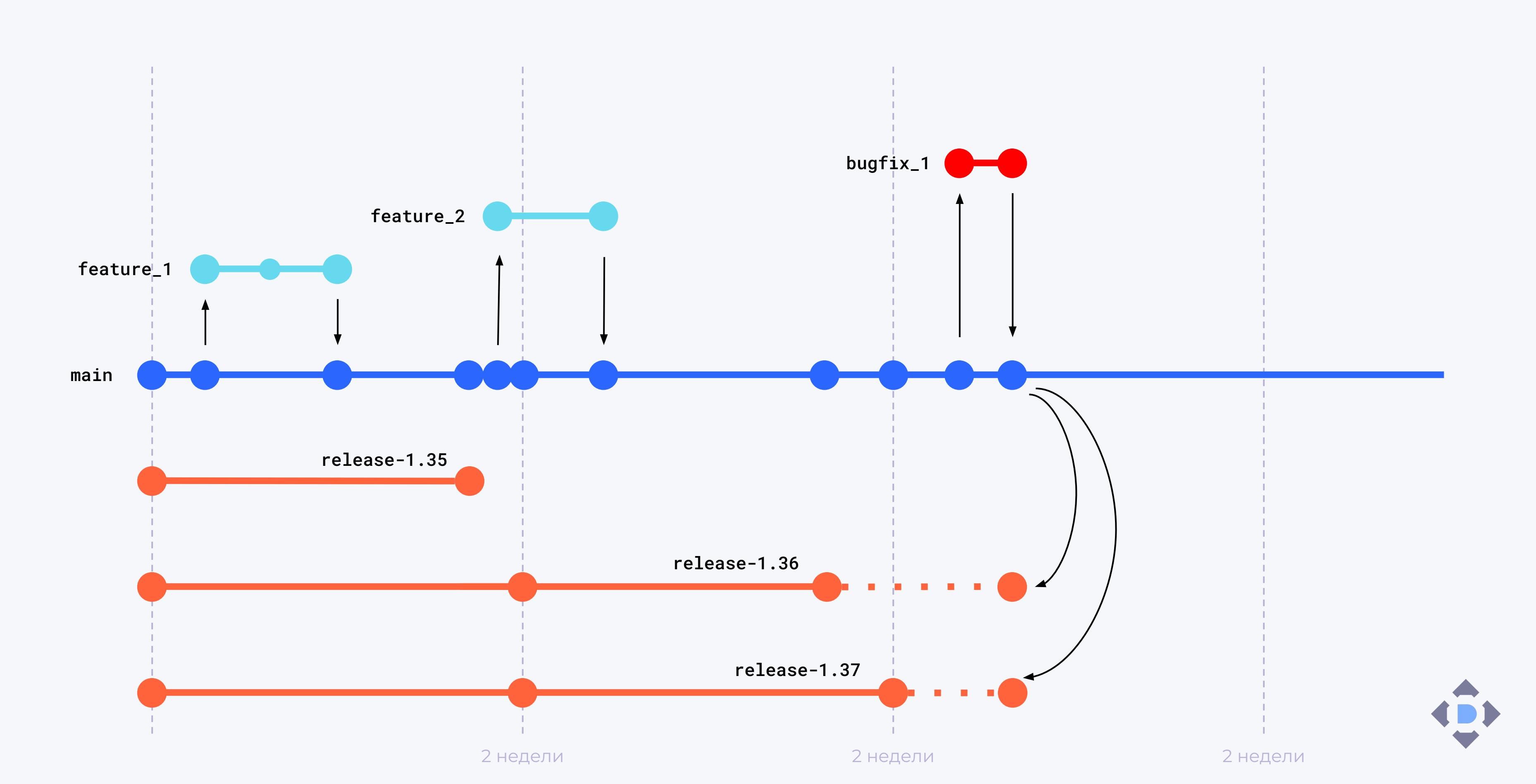
Планирование: release flow



Планирование: release flow

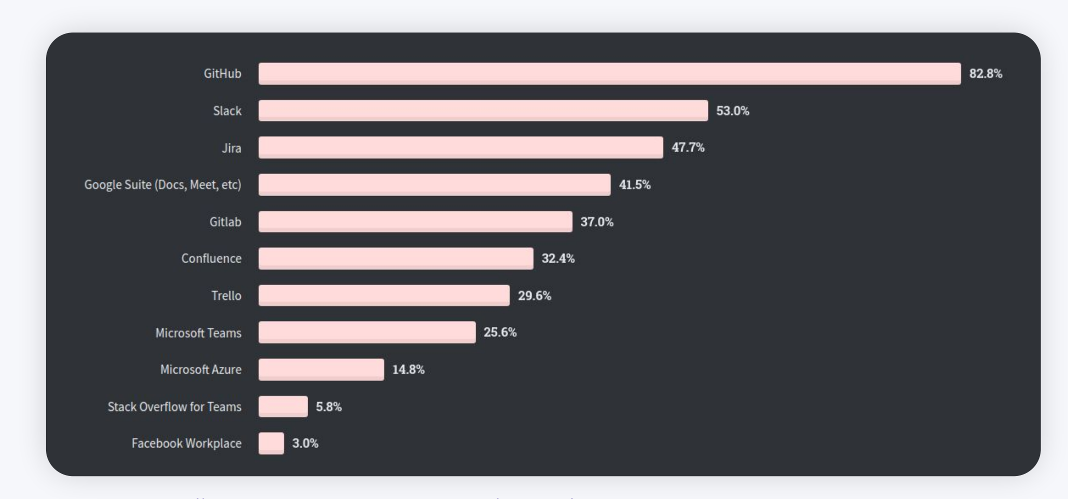


Планирование: release flow

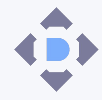


Разработка: GitHub

- Переезжали из standalone GitLab
- Настрадались от GitHub Actions
- Но GitHub по-прежнему остаётся самым популярным



https://insights.stackoverflow.com/survey/2020#technology-collaboration-tools



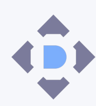
Разработка: Open Source

- Публичные issue и changelog — доверие и уверенность
- Открытые исходники — прозрачность

Deckhouse roadmap

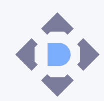
Roadmap by quarters | + New view

	Title	Assignees	Status	Quarter	Milestone	Labels
	2022 Q4 13					
1	[argocd] New module #313	shvgn	In Progress	2022 Q4		area/cicd source/internal-client type/feature-request
2	[opa] Pod Security Standards #2425	yalosev	Done	2022 Q4	v1.38.0	e2e/use/cri/containerd e2e/use/k8s/1.21 skip/documentation-validation ski
3	[virtualization] Workflow - phase 1 of 2 #602	kvaps	In Progress	2022 Q4		area/virtualization type/epic type/feature-request
4	[deckhouse] Release process polishing #1941	diafour, name212…	In Progress	2022 Q4		type/epic
5	E2E tests improvements #1628	apolovov, name2…	In Progress	2022 Q4		type/epic
6	Provide instructions on how to safely migrate workloads from the docker to containerd CR #1420	vitaliy-sn	In Progress	2022 Q4		type/epic type/good-first-issue
7	Go hooks improvements #804	nabokihms	In Progress	2022 Q4		type/epic
8	[deckhouse-config] new module: use DeckhouseConfig custom resource for configuration #1729	diafour	In Progress	2022 Q4	v1.41.0	area/core area/dx e2e/use/cri/containerd e2e/use/k8s/1.21 type/epic
9	[descheduler] Policies for node groups #1382	zuzzas	In Progress	2022 Q4		area/cluster-and-infrastructure area/ux priority/top type/epic type/featu
10	[getting-started] Use nip.io like service for the "Getting Started" instructions #630	z9r5	In Progress	2022 Q4		area/docs priority/top status/no-stale
11	[virtualization] New module #1357	kvaps	In Progress	2022 Q4		area/virtualization
12	[node-manager] Update autoscaler #2735	yalosev	Done	2022 Q4	v1.39.0	e2e/use/cri/containerd e2e/use/k8s/1.21
13	Add automated periodic trivy tests. #2679	gmaiorov	In Progress	2022 Q4		area/compliance area/core area/security area/testing priority/top sou



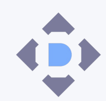
Разработка: идеология Deckhouse

- Минимум «крутилок»
- Собственные интерфейсы, чтобы скрыть реализацию
- NoOps — безболезненные обновления платформы
- Сохраняем «ванильность» K8s, никаких форков



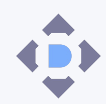
Разработка: идеология Deckhouse

- Минимум «крутилок»
- Собственные интерфейсы, чтобы скрыть реализацию
- NoOps — безболезненные обновления платформы
- Сохраняем «ванильность» K8s, никаких форков



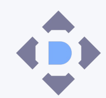
Разработка: идеология Deckhouse

- Минимум «крутилок»
- Собственные интерфейсы, чтобы скрыть реализацию
- NoOps — безболезненные обновления платформы
- Сохраняем «ванильность» K8s, никаких форков



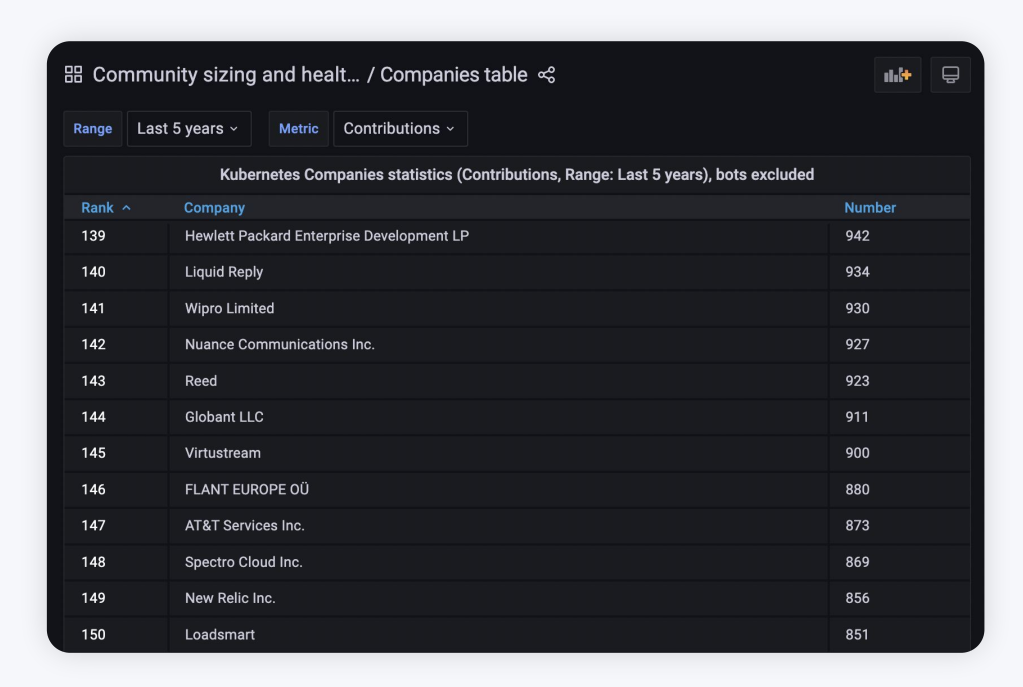
Разработка: идеология Deckhouse

- Минимум «крутилок»
- Собственные интерфейсы, чтобы скрыть реализацию
- NoOps — безболезненные обновления платформы
- Сохраняем «ванильность» K8s, никаких форков



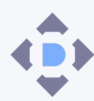
Разработка: наш вклад в Kubernetes

- Топ-1 в России
- Входим в топ-150 в мире


Community sizing and healt... / Companies table
Range: Last 5 years
Metric: Contributions

Kubernetes Companies statistics (Contributions, Range: Last 5 years), bots excluded

Rank	Company	Number
139	Hewlett Packard Enterprise Development LP	942
140	Liquid Reply	934
141	Wipro Limited	930
142	Nuance Communications Inc.	927
143	Reed	923
144	Globant LLC	911
145	Virtustream	900
146	FLANT EUROPE OÜ	880
147	AT&T Services Inc.	873
148	Spectro Cloud Inc.	869
149	New Relic Inc.	856
150	Loadsmart	851



Разработка: наш вклад в Open Source















main · **deckhouse** / modules / 101-cert-manager / images / cert-manager-controller / **patches** /

Go to file · Add file · ...

yalosev fix secret owner ref (#1985) ... ✓ f0b86bf on Jul 8 · **History**

..

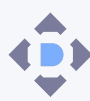
README.md	[cert-manager] Added owner ref certificate parameter (#1601)	6 months ago
certificate_owner_ref.patch	fix secret owner ref (#1985)	4 months ago

README.md

Patches

Certificate owner ref

Adds `CertificateOwnerRef` flag to Certificate CRD. `CertificateOwnerRef` flag is whether to set the certificate resource as an owner of a secret where the TLS certificate is stored. When this flag is enabled, the secret will be automatically removed when the certificate resource is deleted. cert-manager/cert-manager#5158



Разработка: наш вклад в Open Source





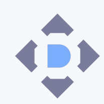









- Все патчи должны быть донесены в upstream
  - Если изменения приняли в upstream — это гарантия, что они будут работать
  - Влияем на развитие важных для нас компонентов
  - Дополнительная оценка наших изменений



Разработка: наш вклад в Open Source





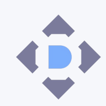






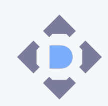



- Все патчи должны быть донесены в upstream
  - Если изменения приняли в upstream — это гарантия, что они будут работать
  - Влияем на развитие важных для нас компонентов
  - Дополнительная оценка наших изменений
  - Наш пример с containerd



Тестирование

- Модульные тесты Helm, hook’ов, OpenAPI
- Модульные матричные тесты
- E2E-тесты для каждого провайдера и конфигурации
- E2E-тесты обновления минорной версии Deckhouse
- Линтеры кода, линтеры шаблонов, валидаторы и прочее



Тестирование: множество провайдеров

- Облачные платформы



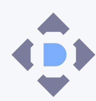





- Другие варианты

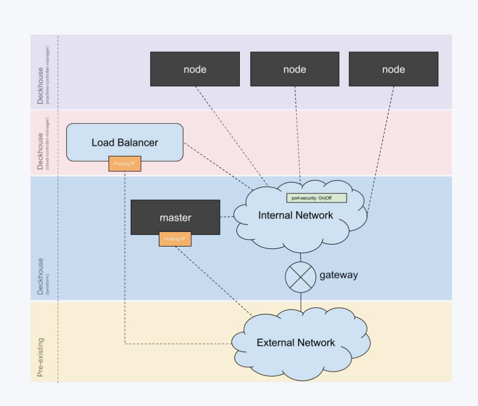




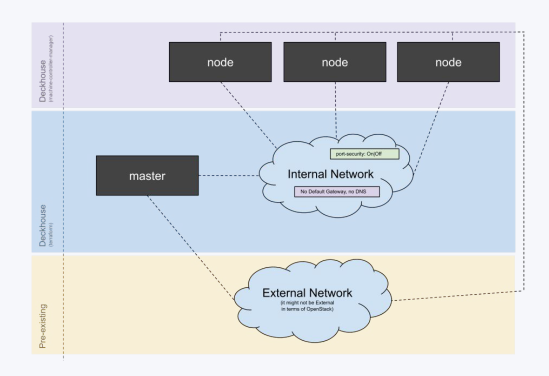


Тестирование: разные схемы размещения

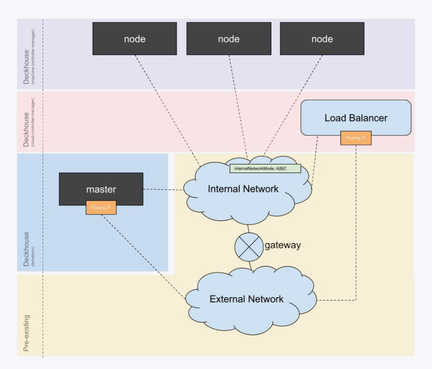
- Стандарт

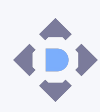


- Без NAT



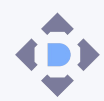
- В уже существующей сети





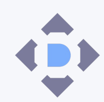
Тестирование: множество вариантов ОС и версий Kubernetes

- РЕД ОС
  - Версии: **7.1**
- AlterOS
  - Версии: **7**
- Astra Linux Special Edition
  - Версии: **1.7**
- Kubernetes
  - Версии: **1.20**, **1.21**, **1.22**, **1.23**, **1.24**
- CentOS
  - Версии: **7**, **8**, **9**
- Debian
  - Версии: **9**, **10**, **11**
- Ubuntu
  - Версии: **16.04**, **18.04**, **20.04**, **22.04**



Тестирование: множество вариантов всего

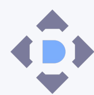
- Поддержка 2-х CRI: Docker и containerd
- Поддержка 2-х CNI: flannel и Cilium
- Несколько десятков модулей



Тестирование: множество вариантов всего

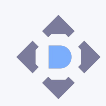
- Поддержка 2-х CRI: Docker и containerd
- Поддержка 2-х CNI: flannel и Cilium
- Несколько десятков модулей





Разработка и тестирование: важные моменты

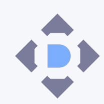
- Публичная разработка дала нам только плюсы
- Следует приносить изменения во все используемые Open Source-проекты
- У продукта должна быть идеология и правила
- Тесты — это хорошо, но всё протестировать невозможно



Разработка и тестирование: важные моменты

- Публичная разработка дала нам только плюсы
- Следует приносить изменения во все используемые Open Source-проекты
- У продукта должна быть идеология и правила
- Тесты — это хорошо, но всё протестировать невозможно

И как же мы решили проблему с тестами?

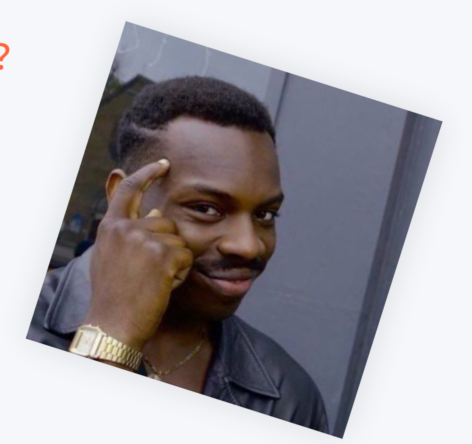


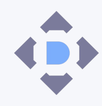
Разработка и тестирование: важные моменты

- Публичная разработка дала нам только плюсы
- Следует приносить изменения во все используемые Open Source-проекты
- У продукта должна быть идеология и правила
- Тесты — это хорошо, но всё протестировать невозможно

И как же мы решили проблему с тестами?

Нужен правильно выстроенный релизный процесс!





Релиз: уверенность

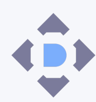

Каналы обновлений


Очередность выката



Окна обновлений


Ручные обновления



Релиз: уверенность



Каналы обновлений



Очередность выката



Окна обновлений

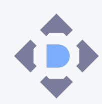


Ручные обновления

Alpha
Beta
Early Access
Stable
Rock Solid

Менее стабильные

Более стабильные



Релиз: уверенность



Каналы обновлений



Очередность выката



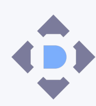
Окна обновлений



Ручные обновления

	ALPHA	BETA	EARLY ACCESS	STABLE	ROCK SOLID
FE	1.40.0 Nov 11, 10:06 Doc · Changelog	1.39.2 Nov 1, 12:07 Doc · Changelog Expect 1.40 on Nov 14 (Mon)	1.39.2 Nov 2, 13:02 Doc · Changelog Expect 1.40 on Nov 16 (Wed)	1.39.2 Nov 8, 13:02 Doc · Changelog Expect 1.40 on Nov 22 (Tue)	1.38.5 Nov 7, 11:42 Doc · Changelog Expect 1.39 on Nov 21 (Mon)
EE	1.39.2 Nov 1, 09:05 Doc · Changelog	1.39.2 Nov 1, 12:07 Doc · Changelog	1.39.2 Nov 2, 13:02 Doc · Changelog	1.39.2 Nov 8, 13:02 Doc · Changelog	1.38.5 Nov 7, 11:42 Doc · Changelog Expect 1.39 on Nov 21 (Mon)
CE	1.39.2 Nov 1, 09:05 Doc · Changelog	1.39.2 Nov 1, 12:07 Doc · Changelog	1.39.2 Nov 2, 13:02 Doc · Changelog	1.39.2 Nov 8, 13:02 Doc · Changelog	1.38.5 Nov 7, 11:42 Doc · Changelog Expect 1.39 on Nov 21 (Mon)

flow.deckhouse.io



Релиз: уверенность



Каналы обновлений



Очередность выката



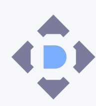
Окна обновлений



Ручные обновления

	ALPHA	BETA	EARLY ACCESS	STABLE	ROCK SOLID
FE	1.40.1 Nov 11, 16:59 Doc · Changelog	1.39.2 Nov 1, 12:07 Doc · Changelog Expect 1.40 on Nov 14 (Mon)	1.39.2 Nov 2, 13:02 Doc · Changelog Expect 1.40 on Nov 16 (Wed)	1.39.2 Nov 8, 13:02 Doc · Changelog Expect 1.40 on Nov 22 (Tue)	1.38.5 Nov 7, 11:42 Doc · Changelog Expect 1.39 on Nov 21 (Mon)
EE	1.40.1 Nov 11, 16:59 Doc · Changelog	1.39.2 Nov 1, 12:07 Doc · Changelog Expect 1.40 on Nov 14 (Mon)	1.39.2 Nov 2, 13:02 Doc · Changelog Expect 1.40 on Nov 16 (Wed)	1.39.2 Nov 8, 13:02 Doc · Changelog Expect 1.40 on Nov 22 (Tue)	1.38.5 Nov 7, 11:42 Doc · Changelog Expect 1.39 on Nov 21 (Mon)
CE	1.40.1 Nov 11, 16:59 Doc · Changelog	1.39.2 Nov 1, 12:07 Doc · Changelog Expect 1.40 on Nov 14 (Mon)	1.39.2 Nov 2, 13:02 Doc · Changelog Expect 1.40 on Nov 16 (Wed)	1.39.2 Nov 8, 13:02 Doc · Changelog Expect 1.40 on Nov 22 (Tue)	1.38.5 Nov 7, 11:42 Doc · Changelog Expect 1.39 on Nov 21 (Mon)

flow.deckhouse.io



Релиз: уверенность


Каналы обновлений

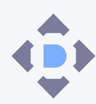

Очередность выката


Окна обновлений


Ручные обновления

	ALPHA	BETA	EARLY ACCESS	STABLE	ROCK SOLID
FE	1.40.2 Nov 15, 14:09 Doc · Changelog	1.40.2 Nov 15, 14:43 Doc · Changelog	1.40.2 Nov 16, 11:36 Doc · Changelog	1.39.2 Nov 8, 13:02 Doc · Changelog Expect 1.40 on Nov 22 (Tue)	1.38.5 Nov 7, 11:42 Doc · Changelog Expect 1.39 on Nov 21 (Mon)
EE	1.40.2 Nov 15, 14:09 Doc · Changelog	1.40.2 Nov 15, 14:43 Doc · Changelog	1.39.2 Nov 2, 13:02 Doc · Changelog Expect 1.40 on Nov 16 (Wed)	1.39.2 Nov 8, 13:02 Doc · Changelog Expect 1.40 on Nov 22 (Tue)	1.38.5 Nov 7, 11:42 Doc · Changelog Expect 1.39 on Nov 21 (Mon)
CE	1.40.2 Nov 15, 14:09 Doc · Changelog	1.40.2 Nov 15, 14:43 Doc · Changelog	1.39.2 Nov 2, 13:02 Doc · Changelog Expect 1.40 on Nov 16 (Wed)	1.39.2 Nov 8, 13:02 Doc · Changelog Expect 1.40 on Nov 22 (Tue)	1.38.5 Nov 7, 11:42 Doc · Changelog Expect 1.39 on Nov 21 (Mon)

flow.deckhouse.io



Релиз: уверенность

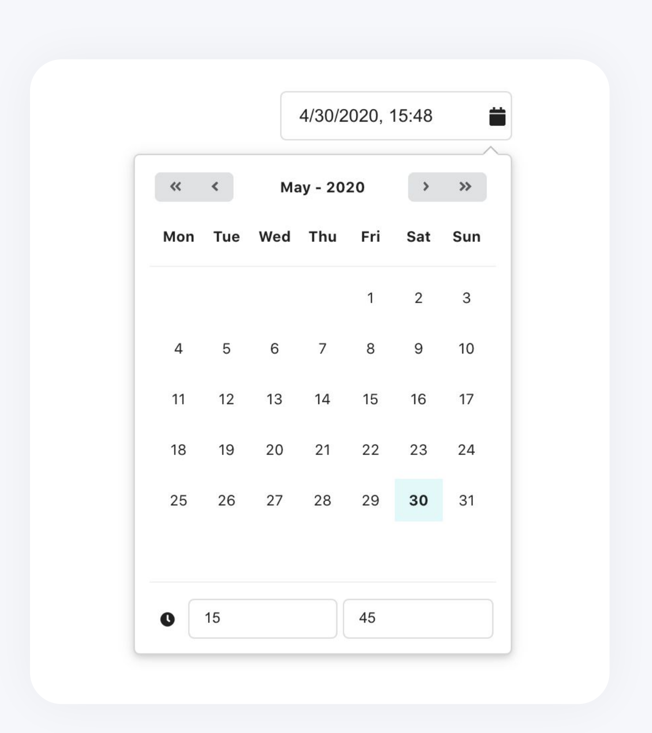

Каналы обновлений

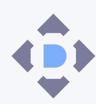

Очередность выката



Окна обновлений


Ручные обновления





Релиз: уверенность



Каналы обновлений



Очередность выката



Окна обновлений

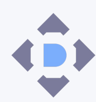


Ручные обновления



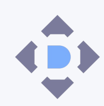
Your Deckhouse is waiting
for the v1.36.5 update approval

```
$ kubectl patch deckhouserelease v1-36-5 \
    --type=merge -p='{"approved": true}'
```



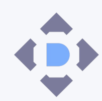
Релиз: люди

- Релиз-инженер
  - Оповещение пользователей
  - Отслеживание процесса релиза (телеметрия и логи)
  - Первичная обработка входящих ошибок по проблемам во время обновления



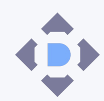
Релиз: люди

- Релиз-инженер
  - Оповещение пользователей
  - Отслеживание процесса релиза (телеметрия и логи)
  - Первичная обработка входящих ошибок по проблемам во время обновления
- Ответственный за релиз
  - Своевременная «заморозка» кодовой базы
  - Проверка changelog’a
  - Проверка E2E по обновлению релиза
  - Финальное согласование выката
  - Поддержка релиз-инженера



Релиз: отслеживание

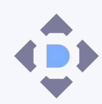
- Логи deckhouse-controller
- Аудит-логи с kube-apiserver
- Телеметрия с информацией о кластере



Релиз: отслеживание

- Логи deckhouse-controller
- Аудит-логи с kube-apiserver
- Телеметрия с информацией о кластере

Всё это собирается только с кластеров под нашим управлением



Релиз: телеметрия

⌄ Table

Clusters

project ↑	cluster_name	pricing_cluster_type	cloud_provider	release_channel	control_plane_version	nodes	state_cluster	state_master	state_terranode
[redacted]	prod	Cloud	vsphere	Stable	1.21	47	ok	ok	ok
	stage	Cloud	vsphere	Alpha	1.21	77	ok	ok	ok
	polk	Cloud	yandex	EarlyAccess	1.21	10	ok	ok	none
	main	Cloud	yandex	Stable	1.21	7	ok	ok	none
	main	Hybrid	openstack	Stable	1.21	31	none	none	none
	adm-production-01-a	Static	none	RockSolid	1.21	23	none	none	none
	stable	Static	none	RockSolid	1.21	28	none	none	none

⌄ Details

Release Channel

current	
Total	236
Stable	107
EarlyAccess	74
Beta	24
RockSolid	11
Unknown	10
Alpha	10

Cloud Provider

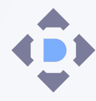
current	
Total	236
none	87
yandex	48
openstack	36
aws	34
vsphere	26
gcp	5

Cluster Type

current	
Total	236
Cloud	149
Static	83
Hybrid	4

Control Plane Version

current	
Total	236
1.21	218
1.23	14
1.22	3



Релиз: телеметрия

CRI / Docker

3500
3000
2500
2000
1500
1000
500
0

08/08 08/16 08/24 09/01 09/08 09/16 09/23 10/01 10/08 10/16 10/24 11/01

	current
Total	2212
19.3.13	2053
18.9.7	100
19.3.15	51
20.10.14	4
18.9.9	4
20.10.17	2
20.10.7	1
18.9.3	1



Релиз: телеметрия

CRI / Containerd

	current
Total	4039
1.5.11-1-gd037943a2	4007
1.4.6	9
1.5.2	8
1.4.11	8
1.5.13	7
1.5.11	2
Unknown	1

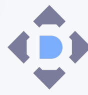
0 500 1000 1500 2000 2500 3000 3500 4000 4500

08/08 08/16 08/24 09/01 09/08 09/16 09/23 10/01 10/08 10/16 10/24 11/01



Релиз: телеметрия

Controller Version Count

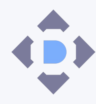


Релиз: процесс обновления

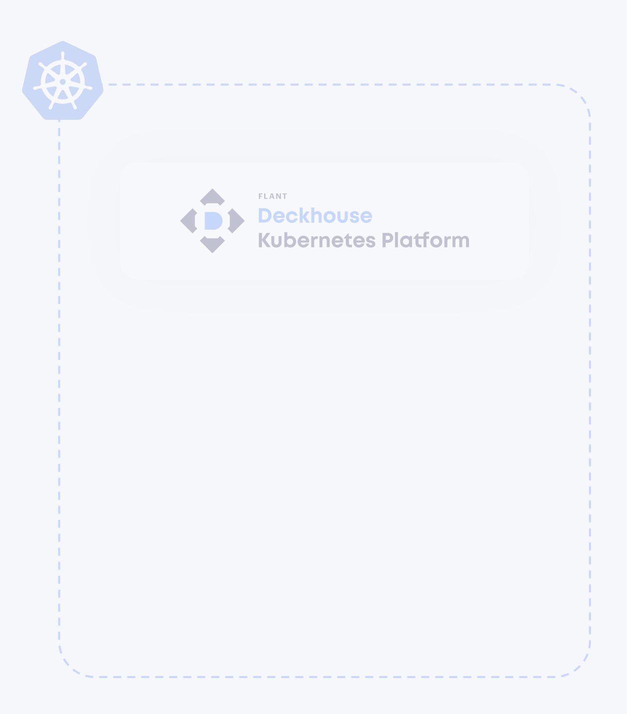




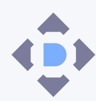
registry.deckhouse.io



Релиз: процесс обновления

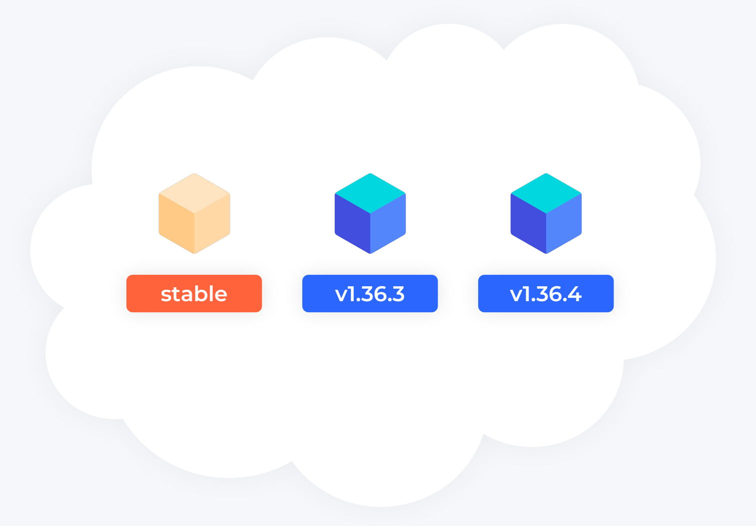




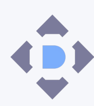


Релиз: процесс обновления





registry.deckhouse.io



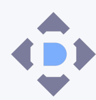
Релиз: процесс обновления

tag: v1.35.9

FLANT
Deckhouse
Kubernetes Platform

stable
v1.36.3
v1.36.4

registry.deckhouse.io



Релиз: процесс обновления

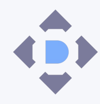
tag: v1.35.9

FLANT
Deckhouse
Kubernetes Platform

Deckhouse Release
v1.36.3

stable
v1.36.3
v1.36.4

registry.deckhouse.io



Релиз: процесс обновления

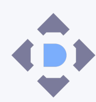
tag: v1.35.9

FLANT
Deckhouse
Kubernetes Platform

Deckhouse Release
v1.36.3
✅ Проверка требований

stable
v1.36.3
v1.36.4

registry.deckhouse.io



Релиз: процесс обновления

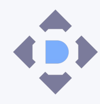
tag: v1.36.3

FLANT
Deckhouse
Kubernetes Platform

Deckhouse Release
v1.36.3

stable
v1.36.3
v1.36.4

registry.deckhouse.io



Релиз: процесс обновления

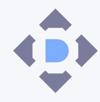
tag: v1.36.3

FLANT
Deckhouse
Kubernetes Platform

Deckhouse Release
v1.36.3

stable
v1.36.3
v1.36.4

registry.deckhouse.io



Релиз: процесс обновления

tag: v1.36.3

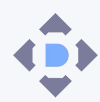
FLANT
Deckhouse
Kubernetes Platform

Deckhouse Release
v1.36.3

Deckhouse Release
v1.36.4

stable
v1.36.3
v1.36.4

registry.deckhouse.io



Релиз: процесс обновления

tag: v1.36.4

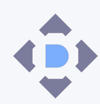
FLANT
Deckhouse
Kubernetes Platform

Deckhouse Release
v1.36.3

Deckhouse Release
v1.36.4

stable
v1.36.3
v1.36.4

registry.deckhouse.io



Релиз: процесс обновления

tag: v1.36.4

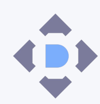
FLANT
Deckhouse
Kubernetes Platform

Deckhouse Release
v1.36.3

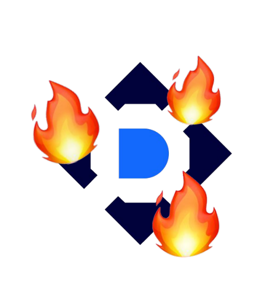
Deckhouse Release
v1.36.4

stable
v1.36.3
v1.36.4

registry.deckhouse.io



Релиз: интересные случаи



Observed
a panic



Проблема:

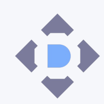
Сломался механизм обновления Deckhouse controller



Решение:

Помогли каналы обновлений и FE-редакция

Сделали E2E для проверки обновлений релизов



Релиз: интересные случаи





Проблема:

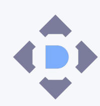
При обновлении kubelet зависали узлы с Docker 18.09



Решение:

Остановили канареечный релиз на Early Access

Прекратили поддержку Docker 18.09



Релиз: интересные случаи



Deprecated
API



Проблема:

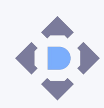
Пользователям пришлось дважды править устаревшие API в одном репозитории



Решение:

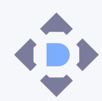
~~10 ударов плетью для тимлида команды разработки~~

Надо обязательно ставить себя на место пользователя



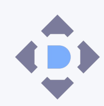
Релиз: важные моменты

- Ретроспектива релиза
- Главное — делать выводы из ошибок
- Нельзя катать по ночам :)



Эксплуатация

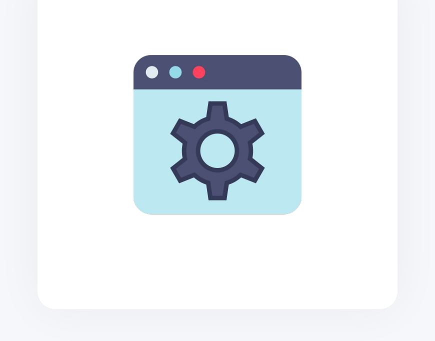
- Большая сервисная история
- Дежурный инженер, дежурный по установке кластеров
- Целая команда разработчиков on-call для решения клиентских проблем
- А ещё есть постоянные демо, пилоты и внедрения



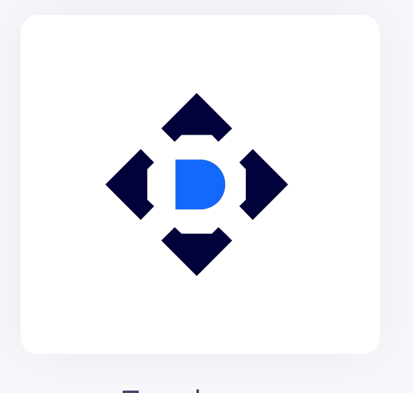
Эксплуатация: внимание к проблемам на стыке



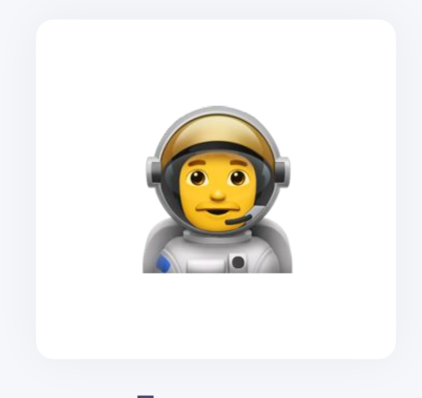
Пользователи



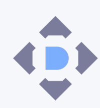
Приложение



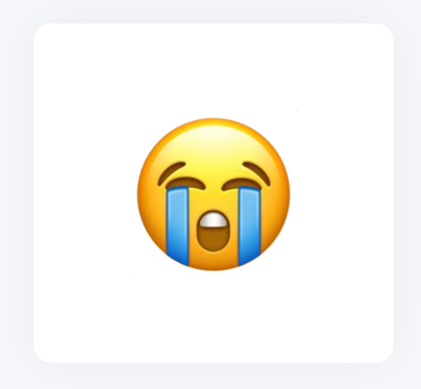
Платформа



Поддержка



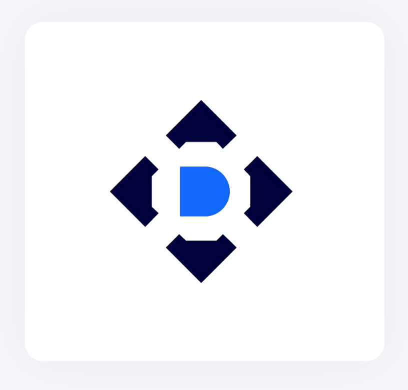
Эксплуатация: внимание к проблемам на стыке



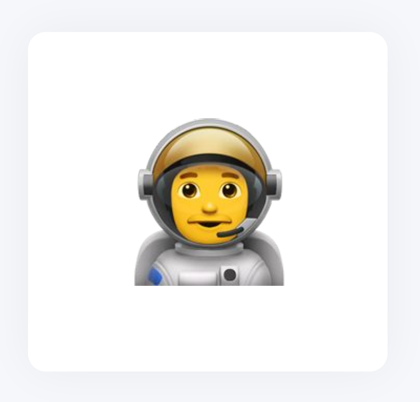

Пользователи



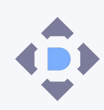
Приложение



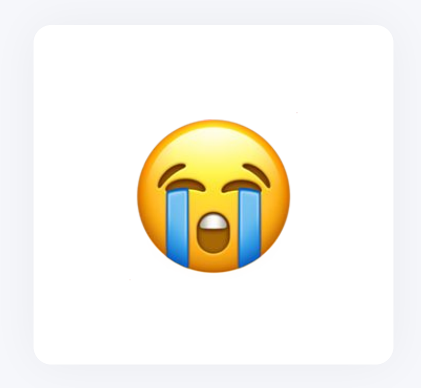
Платформа



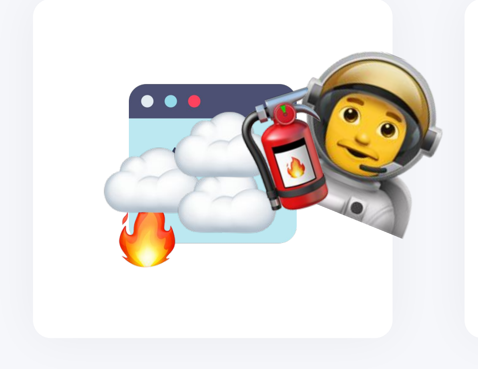
Поддержка



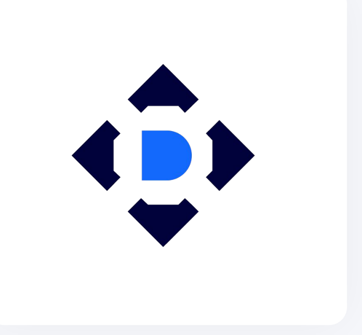
Эксплуатация: внимание к проблемам на стыке



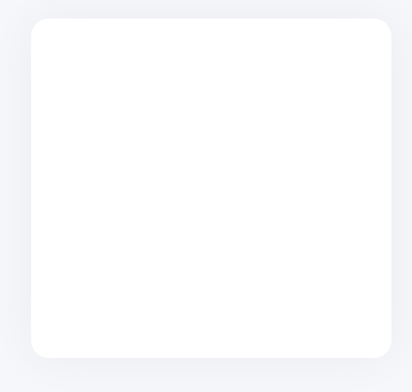
Пользователи



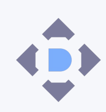
Приложение



Платформа



Поддержка



Эксплуатация: внимание к проблемам на стыке

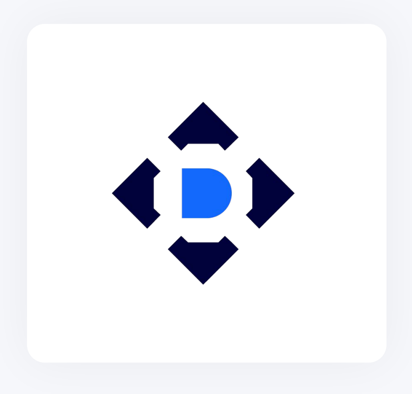




Пользователи



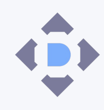
Приложение



Платформа

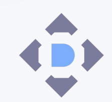


Поддержка



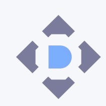
Эксплуатация: сервис

Пришел инженер и в live-режиме решил проблему. Спасибо, вы нас спасли.



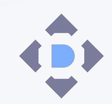
Итоги

- У продукта должна быть идеология и правила
- Протестировать всё невозможно, но правильные процессы закрывают проблему
- Релизы нужно обязательно катать часто, но безопасно и незаметно
- Клиенты ценят сервис
- Обязательно нужно проводить ретроспективу и анализировать ошибки



«Работать с Kubernetes умеет каждый»

Что вы, вообще, сделали?
Взяли Kubernetes и засунули в него разные Open Source-компоненты?

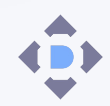


«Работать с Kubernetes умеет каждый»



Что вы, вообще, сделали?
Взяли Kubernetes и засунули в него разные Open Source-компоненты?

MYTH BUSTED





FLANT

Deckhouse Kubernetes Platform

deckhouse.io



Флант
flant.ru



GitHub
github.com/deckhouse



Блог
habr.com/company/flant



Вакансии
job.flant.ru



Константин Аксёнов
konstantin.aksenov@flant.com



Обратная связь и комментарии по докладу по ссылке

